

AD-A129 034

MEASUREMENT OF COMPRESSION OF THE MODIFIED READ CODE IN

1/1

R SCHAPHORST ET AL. 01 NOV 82 NCS-TIB-82-7

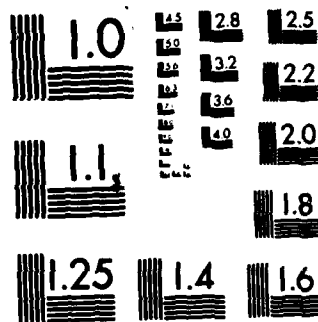
R SCHAPHORST ET AL. 01 NOV 82 NCS-TIB-82-7

UNCLASSIFIED

F/G 9/2

NL

END
DATE
FILMED
7-11-77
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A


NCS TIB 82-7



NATIONAL COMMUNICATIONS SYSTEM

TECHNICAL INFORMATION BULLETIN

82-7

MEASUREMENT OF COMPRESSION OF THE MODIFIED READ CODE II

NOVEMBER 1982

DTIC
ELECTE
JUN 3 1983

S

D

APPROVED FOR PUBLIC RELEASE
DISTRIBUTION UNLIMITED

AD A129034

DTIC FILE COPY

88 06 01 2

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NCS-TIB-82-7	2. GOVT ACCESSION NO. AD A129034	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Measurement of Compression of the Modified READ Code II		5. TYPE OF REPORT & PERIOD COVERED Final Report
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Richard Schaphorst Stephen Urban		8. CONTRACT OR GRANT NUMBER(s) DCA100-80-C-0042
9. PERFORMING ORGANIZATION NAME AND ADDRESS Delta Information Systems 310 Cottman Street Jenkintown, PA 19046		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS National Communications System ATTN: NCS-TS Washington, DC 20305		12. REPORT DATE NOVEMBER 1, 1982
		13. NUMBER OF PAGES 75
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for Public Release; Distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Coding, Compression, Simulation, Facsimile, Modified READ		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The report describes the work on two basic tasks as listed below! <u>Task 1 - Measure Modified READ II Compression</u> It is generally agreed that the Modified READ II code is a prime candidate for use in Group 4 facsimile equipment. The Modified READ II code is the same as the code defined in CCITT Recommendation T.4 with the following exceptions.		

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

- K-Factor 00
- Minimum Scan Line Time 0
- No end of line coded between scan lines
- Two end of line codes at the end of a transmitted page as part of Task 1, 30 computer simulation runs were performed for every combination of the following three parameters.
 - o Resolution (Lines/inch) - 200, 240, 300, 400, 480
 - o CCITT Image No. - 1, 5, 7
 - o Wrap Around - In, Out

Wrap Around refers to the technique where a run length is not necessarily terminated at the end of a scan line. Instead, the coding process is continuous from line to line.

Delta Information Systems measured compression two different ways as listed below:

1. Compression Ratio - Total image pels/Transmitted bits
2. Total transmitted bits

Task 2 - Measure Scan Line Statistics for the Modified READ Code II

Under this task, Delta Information Systems measured the bits/line statistics for the fifteen images processed in Task 1 - 200, 240, 300, 400, and 480 lines/inch; CCITT Images 1, 5, 7. The statistical data includes the minimum bits/line, maximum bits/line, average bits/line, and standard deviation.

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A	

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

NCS TECHNICAL INFORMATION BULLETIN 82-7

MEASUREMENT OF COMPRESSION
OF THE MODIFIED READ CODE II

NOVEMBER 1982

PROJECT OFFICER

DENNIS BODSON
Senior Electronics Engineer
Office of NCS Technology
and Standards

APPROVED FOR PUBLICATION:

Marshall L. Cain

MARSHALL L. CAIN
Assistant Manager
Office of Technology
and Standards

FOREWORD

Among the responsibilities assigned to the Office of the Manager, National Communications System, is the management of the Federal Telecommunication Standards Program. Under this program, the NCS, with the assistance of the Federal Telecommunication Standards Committee identifies, develops, and coordinates proposed Federal Standards which either contribute to the interoperability of functionally similar Federal telecommunication systems or to the achievement of a compatible and efficient interface between computer and telecommunication systems. In developing and coordinating these standards a considerable amount of effort is expended in initiating and pursuing joint standards development efforts with appropriate technical committees of the Electronic Industries Association, the American National Standards Institute, the International Organization for Standardization, and the International Telegraph and Telephone Consultative Committee of the International Telecommunication Union. This Technical Information Bulletin presents an overview of an effort which is contributing to the development of compatible Federal, national, and international standards in the area of digital facsimile standards. It has been prepared to inform interested Federal activities of the progress of these efforts. Any comments, inputs or statements of requirements which could assist in the advancement of this work are welcome and should be addressed to:

Office of the Manager
National Communications System
ATTN: NCS-TS
Washington, D.C. 20305
(202) 692-2124



DELTA INFORMATION SYSTEMS, INC.

310 COTTMAN STREET JENKINTOWN, PA 19046
(215) 572-0840

MEASUREMENT OF COMPRESSION
OF THE MODIFIED READ CODE II

FINAL REPORT

November 1, 1982

Modification P0006 to
Contract No. DCA100-80-C-0042

Submitted to:
NATIONAL COMMUNICATIONS SYSTEM
OFFICE OF TECHNOLOGY AND STANDARDS
Washington, D.C. 20305

Contracting Agency:
DEFENSE COMMUNICATIONS AGENCY

Submitted by:
DELTA INFORMATION SYSTEMS, INC.

310 Cottman Street
Jenkintown, Pa. 19046

MEASUREMENT OF COMPRESSION OF THE
MODIFIED READ CODE II

TABLE OF CONTENTS

	<u>Page</u>
1.0 Introduction	1.0
2.0 Modified READ Code II Algorithm	2.0
3.0 Computer Program Overview	3.0
4.0 Compression Data for the Modified READ Code II	4.0
5.0 Scan Line Statistics for the Modified READ Code II	5.0

APPENDICES

- A. Code Listing for the Program to Simulate Modified READ II
without Wrap-Around
- B. Code Listing for the Program to Simulate Modified READ II
with Wrap-Around

1.0 INTRODUCTION

This document summarizes work performed by Delta Information Systems, Inc. for the office of Technology and Standards of the National Communications System, an organization of the U.S. Government, under Modification P00006 to Contract DCA100-80-C-0042. The Office of Technology and Standards, headed by National Communications System Assistant Manager Marshall L. Cain, is responsible for the management of the Federal Telecommunications Standards Program, which develops telecommunication standards whose use is mandatory by all Federal agencies.

Under the basic contract DCA100-80-C-0042 Delta Information Systems has analyzed alternative resolutions for Group 4 facsimile. The final report for this study was issued in August 1982. As part of this investigation four test documents were scanned with five candidate resolutions, and the results of all twenty scans were printed. The test documents and resolutions used in this study are listed below.

<u>Test Images</u>	<u>Resolutions</u>
CCITT Image No. 1	200 lines/inch
CCITT Image No. 5	240 lines/inch
CCITT Image No. 7	300 lines/inch
Legibility Test Chart	400 lines/inch
	480 lines/inch

After these 20 images were printed it was noted that several pages had artifacts and noisy pels around the edges. These artifacts were removed on subsequent projects in preparation for

the subject simulation study.

The subject project has two basic tasks as listed below.

Task 1 - Measure Modified READ II Compression:

It is generally agreed that the Modified READ II code is a prime candidate for use in Group 4 facsimile equipment. The Modified READ II code is the same as the code defined in CCITT Recommendation T.4 with the following exceptions.

- K-Factor 00
- Minimum Scan Line Time 0
- No end of line codes between scan lines
- Two end of line codes at the end of a transmitted page

As part of Task 1, 30 computer simulation runs were performed for every combination of the following three parameters:

- o Resolution (lines/inch) - 200, 240, 300, 400, 480;
- o CCITT Image No. - 1, 5, 7; and
- o Wrap Around - In, Out.

Wrap Around refers to the technique where a run length is not necessarily terminated at the end of a scan line. Instead, the coding process is continuous from line to line.

Delta Information Systems measured compression two different ways as listed below:

1. Compression Ratio - Total image pels/Transmitted bits,
2. Total transmitted bits

Task 2 - Measure Scan Line Statistics for the Modified READ Code II:

Under this task, Delta Information Systems measured the bits/line statistics for the fifteen images processed in Task 1, and 200, 240,

300, 400, and 480 lines/inch, CCITT Images 1, 5, 7. The statistical data includes the minimum bits/line, maximum bits/line, average bits/line, and standard deviation.

The test results for tasks 1 and 2 are included in sections 4.0 and 5.0 respectively. Section ~~2.0~~² describes the wrap-around algorithm which was simulated while section ~~3.0~~³ provides an overview of the computer program used in the simulation. Appendices A and B contain the code listings for the program to simulate the Mod READ II code with and without wrap-around respectively.

2.0 MODIFIED READ CODE II ALGORITHM

No Wrap-Around

The algorithm for the basic Modified READ Code is defined in CCITT Recommendation T.4 entitled "Standardization of Group 3 Facsimile Apparatus for Document Transmission". It is generally agreed that a strong candidate for the Group 4 coding technique is a variation of the Group 3 Modified READ Code. Table 2-1 is a list of the parameters which differ for the two algorithms.

Table 2-1

Comparison of Group 3 Modified READ Code and Potential
Group 4 Modified READ II Code

	Group 3 T.4 <u>Recommendation</u>	Modified READ Code II
Minimum Scan Line Time	20 MS	0 MS
K-Factor @ 200 lpi	4	∞
End of Line Code between Scan Lines	Yes	No
No. of EOL Codes at End of Page	6	2

Note that in the MRCII code EOL codes are used to change from the one-dimensional coding to the two-dimensional coding or vice versa in addition to two EOL's at the end of the page. This means that for the infinite K factor the first two code lines are preceded by EOL's.

Wrap-Around

The Wrap-Around option provides for images to be treated as a continuous pel stream in which the rightmost pel on a line is followed immediately by the leftmost pel on the next scan line. Otherwise the algorithm is the same as that for no wrap-around. Both the reference line and the coding line continue into the following line with no artificial transition added at line boundaries. This procedure is repeated until end-of-page is reached. That is, if the reference line is n and the coding line is $n+1$, then the reference line is continued into $n+1$ and the coding line is continued into $n+2$. One consequence of this procedure is that long runs in horizontal mode are generated for a sequence of all one-color lines.

3.0 COMPUTER PROGRAM OVERVIEW

The computer programs used to measure the compression of the Mod READ II code are described in this section. Two different programs have been written, one for coding without wrap-around and the other with wrap-around. Code listings of these programs appear in appendices.

3.1 Computer Program without Wrap-Around

This program simulates both the encode and decode processes. To initiate the simulation process, the operator must type in a set of input parameters. The insertion of the input parameters is accomplished on an interactive basis with prompting. After the data has been entered and the measurement parameters have been selected, the first step in the simulation process is the "ENCODE" function. This function detects color changes in the input data and constructs the appropriate code word by table look-up or algorithm. The encoded signal is reversed and fed to the decode function. The decoder basically performs the inverse function of the encoder, generating a series of lines of image pels. The simulation process provides a printout of all the computed performance data as well as a summary tabulation of the input parameters.

The following section describes the structure of the computer program written to simulate the MRCII code. A brief description of each of the functions/subroutines follows:

MRC2

The MRC2 program controls the decoding process. The simulation process is "decode driven;" that is, the main program controls the decode process which decodes a buffered line of compressed data. When the contents of the buffer have been used up, a new line of data is encoded. The MRC2 program also controls parameter input via the INIT routines, and reports computed results.

INIT2

The INIT2 subroutine controls parameter input interactively, prints a summary of the input parameters, and initializes variables.

GETL2

The GETL2 subroutine retrieves a number of requested bits from the coded line and delivers the bits packed into a word (right justified). End-of-line codes (EOL) are detected. If the number of coded bits requested by the calling program is not available, the ENCD2 subroutine is called to provide them.

ENCD2

This subroutine supplies a line of compressed data. Color transitions on an input line are detected bit-by-bit. Both one-dimensional and two-dimensional lines are encoded depending on the parameter K. The code word is generated by table look-up, or algorithm, as appropriate, and added to the coded line buffer via CODELN and/or CODENG.

CODELN

The subroutine CODELN is called by ENCD2 to look up the

Modified Huffman Code (MHC) corresponding to a given run length and color, and add the code word to the coded line buffer.

CODENG

The subroutine CODENG performs a similar function for the two-dimensional case. Based on a particular feature, the appropriate code word is generated by table look-up or algorithm and added to the coded line buffer. All code tables for both one-dimensional and two-dimensional codes are stored in labelled common which is initialized by a BLOCK DATA subprogram.

ONED2

The ONED2 subroutine decodes the MHC. It extracts a set of n bits ($n=3$ initially) from the coded line and looks for a match with all code words of length n , increasing n until a match is found or the code table is exhausted. When and if a match is found, the indicated bits are constructed on the output line.

TWOD2

This subroutine performs the same function as ONED2 for the two-dimensional line.

MI2B and I4B

The subprograms MI2B and I4B are used to pack and unpack a set of bits into (or from) an array of words.

3.2 Computer Program with Wrap-Around

Since the Wrap-Around algorithm requires information from many input lines at the same time, the structure of the program described in the previous section was not suitable for modification to accomodate wrap-around. Also, since the programs are designed to operate in an error free environment, there is no need to inject errors or decode the coded data in order to measure compression. Therefore a new program was written to accommodate wrap-around efficiently, without the decode function. The modules included in this program are described below:

MRCW

The main program, MRCW, simply calls the initialization and encoding routines and reports the results.

ENCDW

This routine performs the one-dimensional and two-dimensional algorithms as required and builds the resultant code lines. Input to this routine is obtained from a list of black-white or white black transitions stored in a circular list in memory.

GTRAN

The subroutine GTRAN scans the input image data for transition and adds the next transition encountered along the scan line to the circular list called TRANS. Transitions are added to the bottom of the list by GTRAN and taken from the top of the list by ENCDW. One transition is added by GTRAN each time it is called. Care must be taken to ensure that the list does not go empty (TOP passes BOTTOM) or that the list overflows (BOTTOM passes TOP).

INIT2

This subroutine is used to enter input parameters interactively and to initialize variables. It performs the initial fill of the transition list.

CODNG & CODLN

These subroutines perform the same functions as for the no wrap-around case, except that CODLN has been modified to accomodate very long runlengths.

4.0 COMPRESSION DATA FOR THE MODIFIED READ CODE II

A computer program was written to measure the number of transmitted bits/page for each of the three CCITT test documents (1, 5, 7) at each of five resolutions (200, 240, 300, 400, 480 lpi). The bits/pg was measured both with and without wrap-around as the algorithms are described in section 2.0. The results of these simulation runs are tabulated in table 4-1. The number of bits/page and the compression ratio is listed for each combination of test document and resolution. The average for the three test CCITT images is also provided. The compression ratio is computed by dividing the number of pels in each test image by the number of compressed bits/pg. The number of pels in each image is listed below.

<u>Resolution</u>	<u>Pels/line</u>	<u>Lines/image</u>	<u>Pels/image</u>
200	1,728	2,336	4,036,608
240	2,048	2,800	5,734,400
300	2,560	3,500	8,960,000
400	3,456	4,672	16,146,432
480	4,096	5,600	22,937,600

Figures 4-1 and 4-2 are graphs of the data in table 4-1 (without wrap-around) for the compressed bits/page and the compression ratio respectively. Note that the compression ratio increases linearly as a function of resolution. Also note that the compression ratio approximately doubles when the resolution doubles. This causes the number of compressed bits/pg to also double as the resolution doubles since the number of pels/image

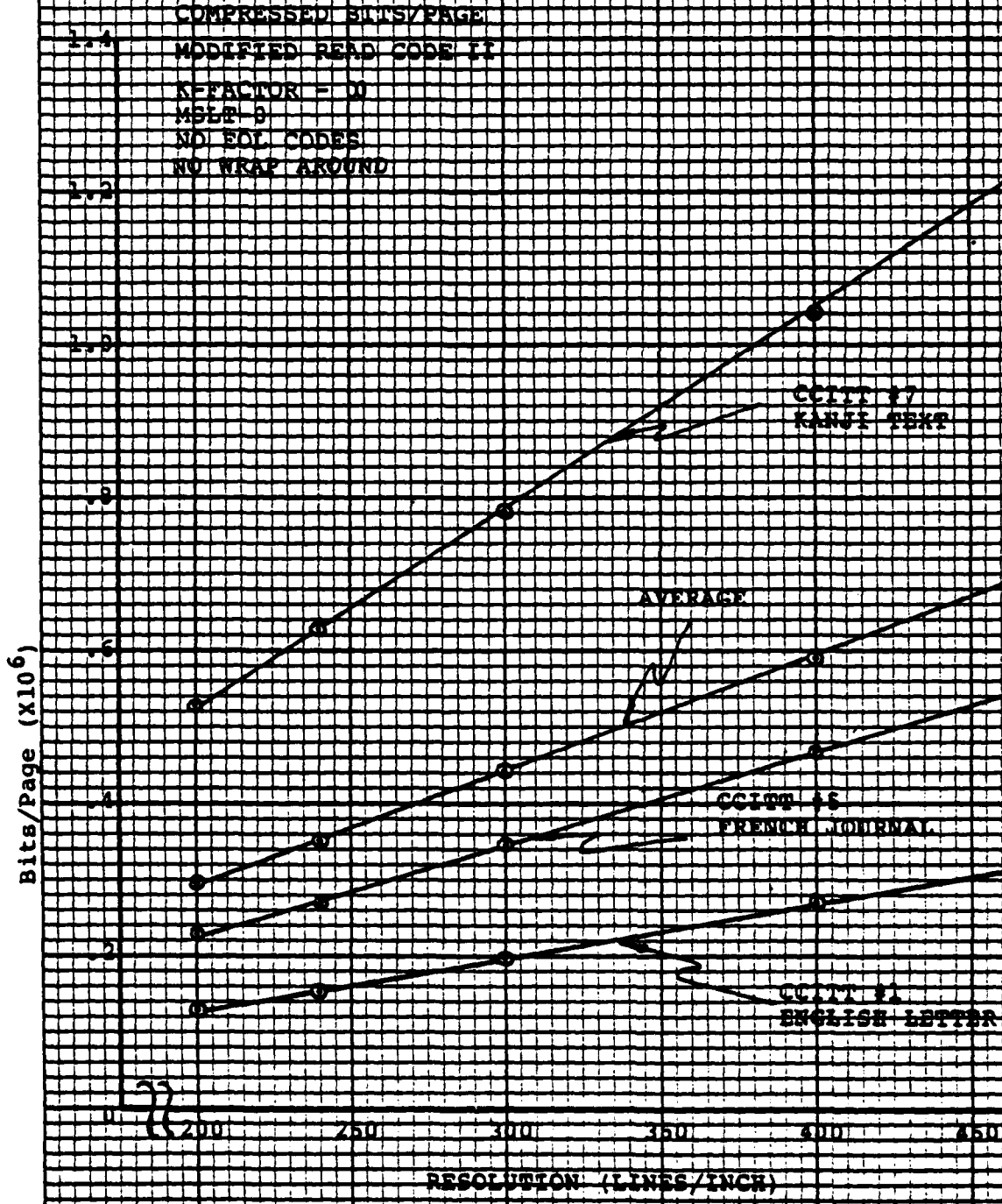
Table 4-1
COMPRESSION DATA FOR THE
MODIFIED READ CODE II

CCITT IMAGE	RESO- LUTION LPI	NO WRAP AROUND		WITH WRAP AROUND		DIFF.
		BITS	COMP. RATIO	BITS	COMP. RATIO	
#1 ENGLISH LETTER	200	132,057	30.57	140,215	28.79	5.8
	240	156,932	36.54	170,073	33.72	7.7
	300	197,170	45.44	218,763	40.96	9.9
	400	271,040	59.57	308,496	52.34	12.1
	480	326,525	70.25	386,287	59.39	15.4
#5 FRENCH JOURNAL	200	229,256	17.61	230,370	17.52	.51
	240	273,023	21.00	274,976	20.85	.71
	300	345,827	25.91	349,976	25.60	1.1
	400	467,330	34.55	476,159	33.91	1.8
	480	570,354	40.22	583,568	39.31	2.2
#7 KANJI	200	531,782	7.59	534,039	7.56	0.4
	240	628,491	9.12	631,118	9.09	0.3
	300	783,644	11.43	788,697	11.36	0.6
	400	1,041,310	15.50	1,050,767	15.37	0.8
	480	1,262,786	18.16	1,276,678	17.97	1.0
AVERAGE	200	297,769	13.56	301,541	13.39	1.2
	240	352,815	16.25	358,722	15.99	1.6
	300	442,217	20.26	452,479	19.80	2.3
	400	593,226	27.22	611,807	26.39	3.0
	480	719,888	31.86	748844	30.63	3.9

ADVANCE
MADE IN USA

GRAPHING PAPER NO. 1240-10
TRACING PAPER NO. 1247-10
CROSS SECTION: 10000 TO 1 INCH

FIGURE 4-1

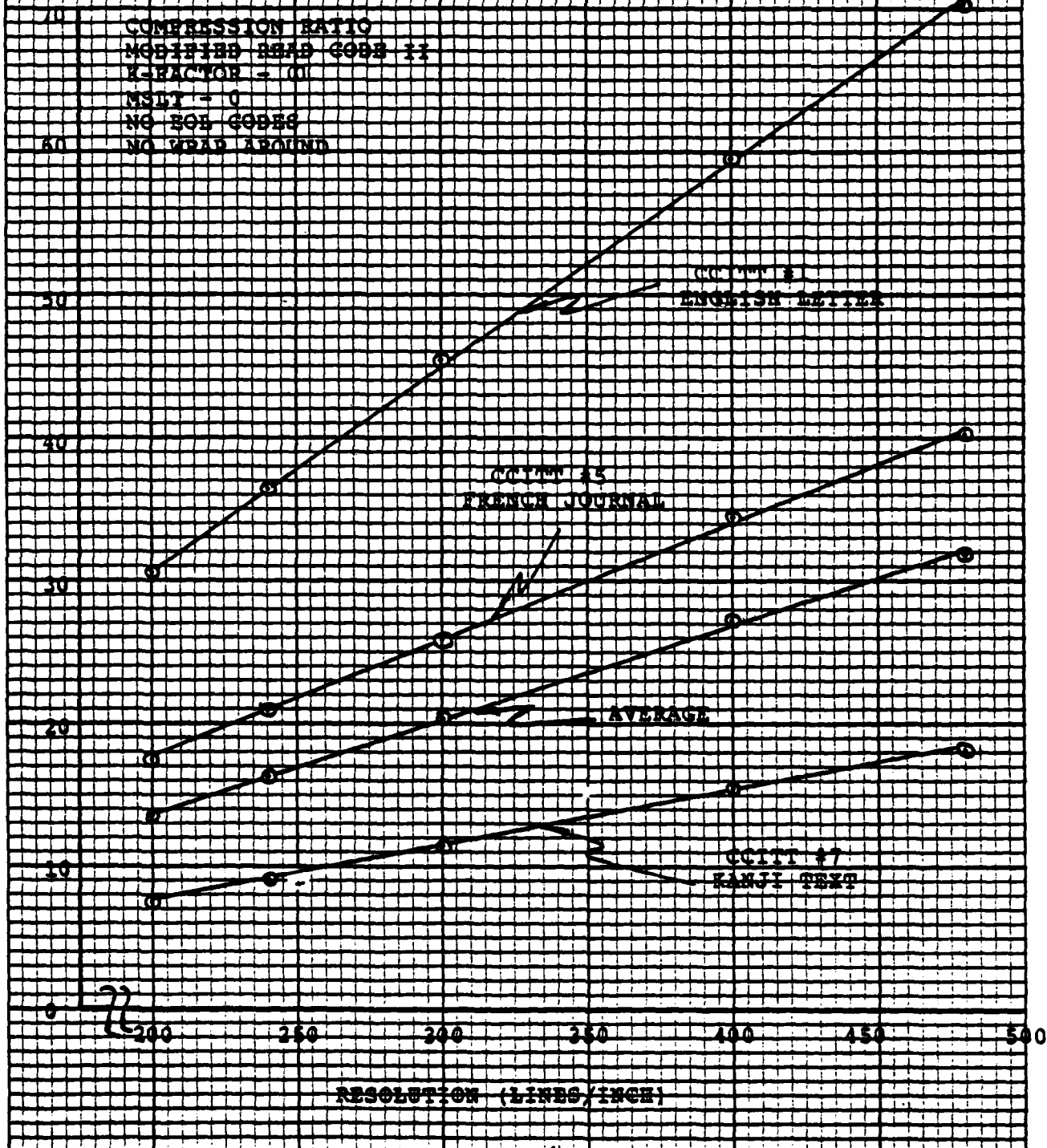


ADAMSEE
MADE IN USA

DRAWING PAPER NO. 1000-10
TRACING PAPER NO. 1007-10
CROSS SECTION-10010 TO 1 INCH

COMPRESSION RATIO

FIGURE 4-3



increases by 4 to 1 as the resolution doubles.

The Wrap Around coding technique was chosen for analysis with the anticipation that it might provide more compression than the non wrap around technique. However, the data in table 4-1 shows that the compression for Wrap Around is consistently poorer than the conventional Mod READ Code II (MRCII) algorithm. The following paragraph examines the Wrap Around technique to give some indication as to why the compression is reduced rather than increased.

The conventional MRCII algorithm transmits an all white line with one bit; this is very efficient. With Wrap Around the average number of bits for a number of consecutive white lines is increased to eight to nineteen per line depending on the resolution. It is this inefficient consecutive all-white-line coding which is primarily responsible for the poorer performance of Wrap Around.

The last column in Table 4-1 is a list of the percentages by which the wrap-around compressions are reduced relative to the non-wrap-around compressions. Notice how the percentage reductions are far greater for the English Letter than they are for the Kanji image. This is due to the fact that there are more all-white lines in the English letter than there are in the Kanji page. Also notice how the percentage generally increases with resolution. This is due to the fact that the number of all-white lines increase with resolution.

This inefficient white-line coding could be improved by transmitting the number of extended runlengths of 2560 pels instead of repeating the code for 2560. However that would require a change to the MRC code table.

5.0 SCAN LINE STATISTICS FOR THE MODIFIED READ CODE II

The computer program which was written to measure the compression of the Modified READ Code II also measured the statistics for the number of bits per scan line for each of the test documents. The following bits/line statistics were measured for each of the test documents

- o minimum bits/line
- o maximum bits/line
- o average bits/line
- o standard deviation

Table 5-1 is a tabulation of these statistics for each test document scanned at each of the five resolutions. Note that the minimum bits/line for each simulation is 1. This occurs because an all-white line is transmitted with one bit regardless of the number of pels in the line. Also note that the average number of bits/line and the standard deviation is relatively independent of the resolution. This is due to the fact that the transmitted bits per line is largely based upon the number of transitions per lines and the number of transitions per line is independent of resolution.

The maximum number of bits/line varies over a relatively wide range for different resolutions due to the totally different locations of scan lines and sampling pels relative to the image.

TABLE 5-1
SCAN LINE STATISTICS FOR THE
MODIFIED READ CODE II*

CCITT IMAGE	RESO- LUTION LPI	MINIMUM BITS/ LINE	MAXIMUM BITS/ LINE	AVERAGE BITS/ LINE	STANDARD DEVIATION
#1 ENGLISH LETTER	200	1	641	56.5	110.9
	240	1	594	56.0	110.7
	300	1	760	56.3	118.4
	400	1	680	58.0	115.3
	480	1	659	58.3	117.3
#5 FRENCH JOURNAL	200	1	907	98.1	132.5
	240	1	778	97.5	133.1
	300	1	755	98.8	127.1
	400	1	1124	100.0	138.6
	480	1	1256	101.8	144.0
#7 KANJI	200	1	507	227.6	123.2
	240	1	514	224.4	123.4
	300	1	530	223.9	127.4
	400	1	559	222.9	123.4
	480	1	605	225.5	126.2

* NO WRAP AROUND

APPENDIX A
CODE LISTING FOR THE MODIFIED READ
CODE II WITHOUT WRAP-AROUND

<u>SUBROUTINE</u>	<u>PAGE NO.</u>
1. MRC2	A-1
2. INIT2	A-7
3. ENCD2	A-11
4. GETL2	A-17
5. BLOCK DATA	A-19
6. ONED2	A-25
7. TWOD2	A-28
8. CODNG	A-32
9. CODLN	A-34
10. INOUT	A-36

```

0001      FTN4.L.T.C
0002      PROGRAM MRC2
0003
0004      IMPLICIT INTEGER(A-Z)
0005
0006      ASSUMPTIONS:
0007      MAXIMUM LINE LENGTH=4096
0008      MAXIMUM NUMBER OF LINES=5600
0009      MAXIMUM INPUT RECORD SIZE=256
0010
0011      COMMON VARIABLES - DEFINITION
0012
0013      INLNNO - INPUT LINE NUMBER
0014      OTLNNO - OUTPUT LINE NUMBER
0015      OTELW - NO. WORDS IN OUTPUT LINE
0016      INELP - NOT USED
0017      CDELP - CODE LINE ELEMENT POINTER
0018      OTELP - OUTPUT LINE ELEMENT POINTER
0019      CDELV - WORDS CONTAINING CODED DATA BITS
0020      CDDATA - NO. OF CODED BITS ON A LINE
0021      CDELC - NO. OF CODED DATA BITS ON A LINE
0022      IMELCT - SPECIFIED BITS ON INPUT LINE
0023      ICDDATA - TOTAL CODED DATA BITS IN IMAGE
0024      TCDEL - TOTAL CODED BITS IN IMAGE
0025      ERREPNT -
0026      ERROFF -
0027      ERPLIM -
0028      ERRCNT - TOTAL ERRORS IN IMAGE
0029      INLNCT - NO. OF INPUT LINES PROCESSED
0030      CONSEC - CONSECUTIVE EOL'S READ
0031      ONECNT - NOT USED
0032      LNKORF - LINE NO. BUFFER
0033      KCHT - NOT USED
0034      ZERO - COUNT OF ZEROS IN CODED LINE
0035      HBPV - NO. OF BITS PER WORD
0036      IMCOD - CODE LINE POINTER (INPUT)
0037      IHREF - REFERENCE LINE POINTER (INPUT)
0038      OTCOD - CODE LINE POINTER (OUTPUT)
0039      OTREF - REFERENCE LINE POINTER (OUTPUT)
0040      STBIT - NO. OF 2-DIMENSIONAL LINES IN IMAGE
0041      ERRCOR -
0042      BUFDM - INPUT RECORD SIZE DESIRED .
0043
0044      C*****LABELLED COMMON /G16BT/ *****
0045
0046      DIMENSION MASK(16),COMASK(16),LIBIT(16),LZBIT(16)
0047      COMMON /G16BT/MASK,COMASK,LIBIT,LZBIT,NBPV
0048
0049      C*****LABELLED COMMON ARRAYS*****
0050
0051      DIMENSION PELREF(258),PELCOD(258),OUTREF(258),OUTCOD(258)
0052      EQUIVALENCE (PELREF,PELBUF),(PELCOD,PELBUF(1,2))
0053      EQUIVALENCE (OUTREF,OTBUF),(OUTCOD,OTBUF(1,2))
0054      COMMON/BUFF/PELBUF(258,2),OTBUF(258,2),
0055      CDBUF(1024),STFBUF(1024),STAT(1)

```

```

0056 COMMON/HUFF/CODE(3,105,2),CODERD(3,11)
0057 COMMON/ERAY/ERRORS(100)
0058
0059 C*****FILE BUFFERS*****
0060 C
0061 COMMON/FILES/TERM,LPFIL,IDC87(144),IDC88(144),IDC89(144)
0062 INTEGER TERM,LPFIL,IDC87,IDC88,IDC89
0063
0064 IDC87 - PELFIL
0065 IDC88 - OTFIL
0066 IDC89 - STATFL
0067 C
0068 C*****LABELLED COMMON VARIABLES*****
0069 C
0070 COMMON/IVAR/PELMAX,VRES,EPHASE,CMPMAX,ERRMOD,LINMAX,K
0071 COMMON/PVAR/INLNNO,OTLNNO,OTELV,CDELP,OTELP,CDELV,CDDATA,
0072 CDELCI,INELCT,TCDATA,TCDEL,ERRPNT,ERROFF,ERRLIM,
0073 CERCNT,INLNCT,CONSEC,LNNOBF,ZERO,
0074 INCOD,INREF,OTCOD,OTREF,STFBIT,ERRCOR,BUFDIM,
0075 COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0076 OUTF
0077 LOGICAL SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0078 OUTF
0079
0080 DOUBLE PRECISION TCDATA,TCDEL
0081 C
0082 C*****END COMMON*****
0083 C
0084 C*****LOCAL VARIABLES*****
0085 C
0086 REAL CF3,CF4,ERRATE
0087
0088 INTEGER STATUS,COLOR,TEMP,BUFDEX,BUFMAX
0089
0090 CALL INIT2
0091 C
0092 C SEARCH MODE: LOOK FOR EOL1 BIT-BY-BIT
0093 C
0094 900 CONTINUE
0095 CALL GETL2(13,MODE,LBITS,L)
0096 GO TO (910,930,930,920),MODE
0097 910 CONTINUE
0098 C
0099 C EOL NOT FOUND; ADVANCE POINTER AND TRY AGAIN
0100 C
0101 CDELP=CDELP+1
0102 GO TO 900
0103 920 CONTINUE
0104 STOP 720
0105 930 CONTINUE
0106 C
0107 C EOL FOUND
0108 C
0109 SEARCH=.FALSE.
0110 CDELP=CDELP+1

```

PAGE 0003 MRC2 5:13 PM SUN., 31 OCT., 1982

```

0111 C
0112 C IF(MODE-2)965,1000,900
0113 C 965 STOP 765
0114 C 1000 CONTINUE
0115 C
0116 C PERFORM ONE-DIMENSIONAL DECODE OF A COMPLETE LINE
0117 C FIRST, SET OUTPUT BUFFER TO WHITE
0118 C (ONLY BLACK RUNS WILL BE INSERTED)
0119 C
0120 C DO 1010 I=1,BUFDIM
0121 C OTBUF(I,OTCOD)=0
0122 C 1010 CONTINUE
0123 C
0124 C INDEX=3
0125 C COLOR=1
0126 C OTELP=1
0127 C
0128 C 1020 CONTINUE
0129 C CALL OMED2(INDEX,COLOR,STATUS,1)
0130 C GO TO (1030,1070,1070,1035,1040),STATUS
0131 C 1 2 3 4 5
0132 C
0133 C RUN ADDED; CHECK LENGTH OF OUTPUT LINE
0134 C
0135 C 1030 CONTINUE
0136 C ONE=TRUE
0137 C IF(OTELP-1-PELMAX) 1031,1032,1050
0138 C 1031 CONTINUE
0139 C IF(COLOR-COLOR=MOD(COLOR+2,2)+1)
0140 C INDEX=3
0141 C GO TO 1020
0142 C 3000 CONTINUE
0143 C
0144 C PERFORM TWO-DIMENSIONAL DECODE
0145 C
0146 C
0147 C FIRST, SET OUTPUT BUFFER TO WHITE
0148 C (ONLY BLACK RUNS WILL BE INSERTED)
0149 C
0150 C DO 3010 I=1,BUFDIM
0151 C OTBUF(I,OTCOD)=0
0152 C 3010 CONTINUE
0153 C
0154 C INDEX=3
0155 C COLOR=1
0156 C OTELP=1
0157 C
0158 C 3020 CONTINUE
0159 C CALL TWOED2(INDEX,COLOR,STATUS,1)
0160 C GO TO (3030,1070,1070,1035,1040),STATUS
0161 C 1 2 3 4 5
0162 C
0163 C RUN ADDED; LOOK FOR NEXT RUN
0164 C
0165 C

```

PAGE 0004 MRC2 5:13 PM SUN., 31 OCT., 1982

```

0166 ONE-.FALSE.
0167 IF(OTELP-1-PELMAX) 3031,1032,1050
0168 3031 CONTINUE
0169 IF(CHCOL)COLOR=MOD(COLOR+2,2)+1
0170 INDEX=3
0171 GO TO 3020
0172 C
0173 C LINE LENGTH=PELMAX; NO FILL EXPECTED
0174 C IF EOL, CONTINUE IN SPECIFIED MODE
0175 C IF NO EOL, CONTINUE IN PREVIOUS MODE
0176 C
0177 1032 CONTINUE
0178 C
0179 C CHECK FOR EOL
0180 C
0181 CALL GETL2(13,MODE,LBITS,L)
0182 GO TO (1065,1060,1060,1050).MODE
0183 C
0184 C PREMATURE EOL DETECTED
0185 C
0186 C
0187 C EOL1 DETECTED
0188 C
0189 1035 CONTINUE
0190 CDELP=CDELP+L
0191 STATUS=4
0192 IF(OTELP.LE.1) CONSEC=CONSEC+1
0193 IF(CONSEC-2)1080,1080,2000
0194 C
0195 C EOL2 DETECTED
0196 C
0197 1040 CONTINUE
0198 CDELP=CDELP+L
0199 STATUS=5
0200 C
0201 C GO TO 1080
0202 C
0203 C PROBLEMS, PROBLEMS
0204 C
0205 1050 STOP 1050
0206 C
0207 C LINE LENGTH CORRECT, WRITE OUTPUT LINE
0208 C
0209 1060 CONTINUE
0210 C
0211 C EOL DETECTED PROPERLY
0212 C
0213 CDELP=CDELP+L
0214 CONSEC=1
0215 IF(ONE) SYNC=TRUE.
0216 C
0217 C ENTRY FOR NO EOL
0218 C
0219 1065 CONTINUE
0220 IEOF=1

```

PAGE 0005 MRC2 5:13 PM SUM., 31 OCT., 1982

```

0221 IF(OUTF) CALL INOUT(OTCOD+2,IEOF)
0222 OTLNO-LMNOBF
0223 TEMP-OTREF
0224 OTREF-OTCOD
0225 OTCOD-TEMP
0226 IF(MODE.EQ.2) GO TO 1000
0227 GO TO 3000
0228 C
0229 C LINE TOO LONG OR NO MATCH
0230 C
0231 1000 CONTINUE
0232 WRITE=.FALSE.
0233 DIAG=.TRUE.
0234 C
0235 C LINE SHORT
0236 C
0237 1000 CONTINUE
0238 IF(STATUS.EQ.4) GO TO 1000
0239 SEARCH=.TRUE.
0240 GO TO 900
0241 C
0242 C END OF MESSAGE
0243 C
0244 2000 CONTINUE
0245 WRITE(LPFILE,2010) CONSEC
0246 2010 FORMAT('END OF MESSAGE DETECTED ("',I2," EOL'S")')
0247 C
0248 C REPORT COMPRESSION FACTOR, ERROR SENSITIVITY FACTOR,BIT ERROR RATE
0249 C
0250 ERRATE=FLOAT(ERRCNT)/TCDEL
0251 WRITE(LPFILE,2020) TCDEL,TCDATA,STEBIT,INLNCT,ERRATE
0252 2020 FORMAT('CODED BITS =',F8.0/
0253 ' - CODED DATA BITS =',F8.0/
0254 ' - 2-DIM LINES =',I8/
0255 ' - INPUT LINES PROCESSED =',I8/
0256 ' - BIT ERROR RATE =',G14.6)
0257 C
0258 CF3=FLOAT(PELMAX)*FLOAT(INLNCT)/TCDEL
0259 CF4=FLOAT(PELMAX)*FLOAT(INLNCT)/TCDATA
0260 C
0261 WRITE(LPFILE,2030) CF3,CF4
0262 2030 FORMAT('B(CF3) =',F8.4/
0263 ' (CF4) =',F8.4)
0264 C
0265 C
0266 C WRITE EOF INDICATOR ON STAT FILE & CLOSE
0267 C
0268 STAT(1)=-1
0269 CALL WRITE(IDC9,IERR,STAT)
0270 IF(IERR.LT.#)STOP 203
0271 CALL CLOSE(IDC9)
0272 IF(OUTF) CALL ERRMS
0273 C
0274 CALL FTIME(PELREF)
0275 WRITE(LPFILE,4000) (PELREF(I),I=1,15)

```

PAGE 0006 MRC2 5:13 PM SUN., 31 OCT., 1982

0276 4000 FORMAT(1H,15A2)
0277 CALL INOUT(99,1EOF)
0278 E N D

FTN4 COMPILER: HP92000-16092 REV. 2026 (800423)

COMMON = 00000

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00577

PAGE 0001 FTM. 5:14 PM SUN., 31 OCT., 1982

```

0001      FTM4,L,T,C
0002      SUBROUTINE INIT2
0003      C
0004      IMPLICIT INTEGER(A-Z)
0005      C
0006      *****LABELLED COMMON /G168T/ *****
0007      C
0008      DIMENSION MASK(16),COMASK(16),LIBIT(16),LZBIT(16)
0009      COMMON /G168T/MASK,COMASK,LIBIT,LZBIT,NBPV
0010      C
0011      *****LABELLED COMMON ARRAYS*****
0012      C
0013      DIMENSION PELREF(258),PELCOD(258),OUTREF(258),OUTCOD(258)
0014      EQUIVALENCE ((PELREF,PELBUF),(PELCOD,PELBUF(1,2)))
0015      EQUIVALENCE ((OUTREF,OTBUF),(OUTCOD,OTBUF(1,2)))
0016      COMMON/BUFF/PELBUF(258,2),OTBUF(258,2),STAT(1)
0017      & CDBUF(1024),STFBUF(1024)
0018      COMMON/HUFF/CODE(3,105,2),CODERD(3,11)
0019      COMMON/ERAY/ERRORS(100)
0020      C
0021      *****FILE BUFFERS*****
0022      C
0023      COMMON/FILES/TERM,LPFIL,IDCB7(144),IDCB8(144),IDCB9(144)
0024      INTEGER TERM,LPFIL,IDCB7,IDCB8,IDCB9
0025      C
0026      IDC87 - PELFIL
0027      IDC88 - OTFIL
0028      IDC89 - STATFL
0029      C
0030      *****LABELLED COMMON VARIABLES*****
0031      C
0032      COMMON/IVAR/PELMAX,VRES,EPHASE,CHPMAX,ERRMOD,LINMAX,K
0033      COMMON/PVAR/INLNNO,OTLNNO,OTELV,COELP,OTELP,CDELV,CDDATA,
0034      CDELCT,INELCT,TCDATA,TCDEL,ERRPNT,ERROFF,ERRLIM,
0035      * ERRCNT,INLMCT,CONSEC,LNNOBF,ZERO,
0036      * INCOD,INREF,OTCOD,OTREF,STFBIT,ERRCOR,BUFDIM
0037      * COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0038      * OUTF
0039      * LOGICAL SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0040      * OUTF
0041      C
0042      DOUBLE PRECISION TCDATA,TCDEL
0043      C
0044      *****END COMMON*****
0045      C
0046      *****FILE PARAMETERS*****
0047      C
0048      DIMENSION LINE(80),IBUF(10),ITBUF(15),NMBUF(3)
0049      C
0050      EQUIVALENCE (IBUF,FNAM), (IBUF(5),ISECU), (IBUF(6),ICR)
0051      C
0052      DATA LINE/80*2H /,LEN/80/,ISTRC/1/
0053      C
0054      *****BEGIN PROGRAM*****
0055      C

```

```

0056 C
0057 C GET INPUT IMAGE FILE NAME AND OPEN
0058 C
0059 CALL GETST(LINE,LEN,IDUM)
0060 CALL NAMR(1BUF,LINE,2*IDUM,ISTR)
0061 CALL OPEN(IDC87,IERR,FNAM,B,ISECU,ICR)
0062 IF(IERR.LT.B)STOP 0001
0063 C
0064 C GET OUTPUT IMAGE FILE NAME AND OPEN
0065 C
0066 CALL NAMR(1BUF,LINE,2*IDUM,ISTR)
0067 CALL OPEN(IDC88,IERR,FNAM,B,ISECU,ICR)
0068 IF(IERR.GE.B) GO TO 5
0069 C
0070 C
0071 C IF NO OUTPUT FILE
0072 C THEN NO ERROR INSERTION
0073 C AND
0074 C NO ERROR COMPARISON (EARNMS)
0075 C
0076 C NO INPUT LINE NUMBER OR PEL COUNT USED/REQUIRED
0077 C ELSE ERROR INSERTION OPTIONAL
0078 C
0079 C
0080 C NO OUTPUT FILE
0081 C
0082 C
0083 C
0084 C
0085 C
0086 C
0087 C
0088 C
0089 C
0090 C
0091 C
0092 C
0093 C
0094 C
0095 C
0096 C
0097 C
0098 C
0099 C
0100 C
0101 C
0102 C
0103 C
0104 C
0105 C
0106 C
0107 C
0108 C
0109 C
0110 C
0111 C
0112 C
0113 C
0114 C
0115 C
0116 C
0117 C
0118 C
0119 C
0120 C
0121 C
0122 C
0123 C
0124 C
0125 C
0126 C
0127 C
0128 C
0129 C
0130 C
0131 C
0132 C
0133 C
0134 C
0135 C
0136 C
0137 C
0138 C
0139 C
0140 C
0141 C
0142 C
0143 C
0144 C
0145 C
0146 C
0147 C
0148 C
0149 C
0150 C
0151 C
0152 C
0153 C
0154 C
0155 C
0156 C
0157 C
0158 C
0159 C
0160 C
0161 C
0162 C
0163 C
0164 C
0165 C
0166 C
0167 C
0168 C
0169 C
0170 C
0171 C
0172 C
0173 C
0174 C
0175 C
0176 C
0177 C
0178 C
0179 C
0180 C
0181 C
0182 C
0183 C
0184 C
0185 C
0186 C
0187 C
0188 C
0189 C
0190 C
0191 C
0192 C
0193 C
0194 C
0195 C
0196 C
0197 C
0198 C
0199 C
0200 C

```

PAGE 8883 INIT2 5:14 PM SUN.. 31 OCT.. 1982

```

0111 116 CONTINUE
0112     DIAG=.TRUE.
0113 C
0114 C   READ MAXIMUM NUMBER OF PELS PER LINE
0115 C
0116 128 CONTINUE
0117     WRITE(TERM,138)
0118 138 FORMAT("ENTER MAXIMUM NUMBER OF PELS PER LINE: ")
0119     READ(TERM,"") PELMAX
0120     148 FORMAT(14)
0121     IF(PELMAX.GE.1.AND.PELMAX.LE.4896) GO TO 168
0122     WRITE(TERM,158) PELMAX
0123 158 FORMAT("NUMBER OUT OF RANGE (-.16.)")
0124     GO TO 128
0125 C
0126 C   READ VERTICAL SAMPLING
0127 C
0128 168 CONTINUE
0129     WRITE(TERM,178)
0130 178 FORMAT("ENTER VERTICAL SAMPLING: ")
0131     READ(TERM,"") VRES
0132     IF(VRES.GE.1.AND.VRES.LE.18) GO TO 198
0133     WRITE(TERM,158) VRES
0134     GO TO 168
0135 C
0136 C   READ PARAMETER K
0137 C
0138 198 CONTINUE
0139     WRITE(TERM,192)
0140 192 FORMAT("ENTER PARAMETER K: ")
0141     READ(TERM,"") K
0142     IF(K.GE.1.AND.K.LE.5688) GO TO 328
0143     WRITE(TERM,158) K
0144     GO TO 198
0145 C
0146 C   READ NUMBER OF SCAN LINES TO BE PROCESSED
0147 C
0148 328 CONTINUE
0149     WRITE(TERM,338)
0150 338 FORMAT("NUMBER OF SCAN LINES TO BE PROCESSED=? ")
0151     READ(TERM,"") LINMAX
0152     IF(LINMAX.GE.1.AND.LINMAX.LE.5688) GO TO 358
0153     WRITE(TERM,158) LINMAX
0154     GO TO 328
0155 358 CONTINUE
0156 C
0157 C   READ INPUT IMAGE NAME
0158 C
0159     WRITE(TERM,368)
0160 368 FORMAT("ENTER INPUT IMAGE NAME:")
0161     365 FORMAT(3A2)
0162     WRITE INPUT PARAMETERS
0163     CALL FTIME(ITBUF)
0164     WRITE(LPFILE,378) ITBUF
0165 378 FORMAT(1H8,15A2)

```

PAGE 0004 INIT2 5:14 PM SUN., 31 OCT., 1982

```

0165 WRITE(LPFIL,300) NMBUF
0167 300 FORMAT('IMAGE NAME - '3A2)
0168 C
0169 WRITE(LPFIL,400) PELMAX,VRES,K,LINMAX,BUFDIM
0170 400 FORMAT('INPUT PARAMETERS: /
0171 - MAXIMUM NUMBER OF PELS PER LINE-',16/
0172 - VERTICAL SAMPLING: N-',14/
0173 - PARAMETER K -',14/
0174 - NUMBER OF SCAN LINES TO BE PROCESSED -',16/
0175 - RECORD SIZE -',14)
0176 WRITE(LPFIL,410)
0177 410 FORMAT ('NO ERRORS INSERTED')
0178 C***** BEGIN PROGRAM *****
0179 C
0180 C INITIALIZE
0181 C
0182 INELCT=PELMAX
0183 CDELT=NSPW
0184 CDELP=NSPW+1
0185 DO 850 I=1,BUFDIM+4
0186 STBUF(I)=0
0187 CDBUF(I)=0
0188 850 CONTINUE
0189 DO 850 I=1,BUFDIM
0190 OTBUF(I,OTREF)=0
0191 OTBUF(I,OTCOD)=0
0192 PELBUF(I,INREF)=0
0193 PELBUF(I,INCOD)=0
0194 850 CONTINUE
0195 RETURN
0196 E N D

```

FTM4 COMPILER: HP92060-16092 REV. 2026 (000423)

** NO WARNINGS ** NO ERRORS ** PROGRAM - 00041 COMMON - 00000

```

8081      FTN4,L,T,C
8082      SUBROUTINE ENCD2
8083      C
8084      C      IMPLICIT INTEGER(A-Z)
8085      C*****LABELLED COMMON /G16BT/ *****
8086      C
8087      C      DIMENSION MASK(16),COMASK(16),LIBIT(16),LZBIT(16)
8088      C      COMMON /G16BT/MASK,COMASK,LIBIT,LZBIT,NBPV
8089      C
8090      C*****LABELLED COMMON ARRAYS*****
8091      C
8092      C      DIMENSION PELREF(258),PELCOD(258),OUTREF(258),OUTCOD(258)
8093      C      EQUIVALENCE (PELREF,PELBUF),(PELCOD,PELBUF(1,2))
8094      C      EQUIVALENCE (OUTREF,OTBUF),(OUTCOD,OTBUF(1,2))
8095      C      COMMON/BUFF/PELBUF(258,2),OTBUF(258,2)
8096      C      CDBUF(1024),STFBUF(1024),STAT(1)
8097      C      COMMON/HUFF/CODE(3,105,2),CODERD(3,11)
8098      C      COMMON/ERAY/ERRORS(100)
8099      C
8100      C*****FILE BUFFERS*****
8101      C
8102      C      COMMON/FILES/TERM,LPFIL,IDC87(144),IDC88(144),IDC89(144)
8103      C      INTEGER TERM,LPFIL,IDC87,IDC88,IDC89
8104      C
8105      C      IDC87 - PELFIL
8106      C      IDC88 - OTFIL
8107      C      IDC89 - STATFL
8108      C
8109      C*****LABELLED COMMON VARIABLES*****
8110      C
8111      C      COMMON/IVAR/PELMAX,VRES,EPHASE,CMPMAX,ERRMOD,LINMAX,K
8112      C      COMMON/PVAR/IMLNNO,OTLNNO,OTELV,CDELP,OTELP,CDELV,CDDATA,
8113      C      CDELCI,INELCT,TCDATA,TCDEL,ERRPNT,ERROFF,ERRLIN,
8114      C      ERRCNT,INLNCT,CONSEC,LMNOBF,ZERO,
8115      C      INCOD,INREF,OTCOD,OTREF,STFBIT,ERRCOR,BUFDIM
8116      C      COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
8117      C      OUTF
8118      C      LOGICAL SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
8119      C      OUTF
8120      C
8121      C      DOUBLE PRECISION TCDATA,TCDEL
8122      C
8123      C*****END COMMON*****
8124      C
8125      C      INTEGER AB,A1,A2,BIT,B1,B2,ERRBIT,FILL,I,MAB,PEL,PELMI
8126      C      INTEGER POL,POLAR,RUN,TEMP
8127      C***** BEGIN PROGRAM *****
8128      C
8129      C      INITIALIZE VARIABLES
8130      C
8131      C      CDELCI=NBPV
8132      C      CDDATA=B
8133      C      DO 50 I=2,BUFDIM*4
8134      C      CDBUF(I)=0
8135      C

```

```

0056 STFBUF(1)=0
0057 50 CONTINUE
0058 C
0059 C READ INPUT PICTURE FILE
0060 C
0061 100 CONTINUE
0062 IF(FILEND) GO TO 120
0063 IF(INLNCT-GE.LINMAX)GO TO 120
0064 CALL INOUT(INCOD,IEOF)
0065 IF(IEOF.EQ.-1) GO TO 120
0066 IF(.NOT.OUTF) INLNNO-INLNNO+1
0067 IF(MOD(INLNNO-1,VRES).NE.0) GO TO 100
0068 IF(INLNCT.LT.PELMAX) STOP 2222
0069 INLNCT-INLNCT+1
0070 C
0071 C LOAD OUTPUT LINE NUMBER BUFFER
0072 C
0073 LNNOF-INLNNO
0074 IF(SEARCH)OTLNNO-LNNOF
0075 C
0076 IF(INLNNO.LE.LINMAX) GO TO 140
0077 C
0078 C WRITE SIX EOL'S
0079 C
0080 120 CONTINUE
0081 FILEND=.TRUE.
0082 DO 130 I=1,6
0083 CALL CODNG(10,0,0,0,0,0)
0084 130 CONTINUE
0085 DO 135 I=1,6
0086 STFBUF(1)=CDBUF(1)
0087 135 CONTINUE
0088 GO TO 300
0089 C
0090 C FIRST OF K LINES
0091 C
0092 140 CONTINUE
0093 IF(MOD(INLNCT-1,K).NE.0) GO TO 600
0094 C
0095 ONE-DIMENSIONAL CODING
0096 C
0097 WRITE ONE EOL
0098 C
0099 CALL CODNG(10,0,0,0,0,0)
0100 C
0101 POLAR=1
0102 C
0103 TEST COLOR OF FIRST ELEMENT
0104 C
0105 IF(140PELBUF(1,INCOD),1,1).EQ.0) GO TO 150
0106 C
0107 FIRST ELEMENT BLACK; ENCODE 0-LENGTH WHITE RUN
0108 C
0109 CALL CODLN(0,1)
0110 POLAR=2
0111 C

```

PAGE 8883 ENC02 5:13 PM SUN.. 31 OCT., 1982

```

0111 C
0112 C CALCULATE RUN LENGTH AND ENCODE
0113 C
0114 15# CONTINUE
0115 RUN-B
0116 DO 2# I=1,PELMAX
0117 PEL=IAB(PELBUF(1,INCODE),1,1)+1
0118 IF(PEL.EQ.POLAR) GO TO 18#
0119 CALL CODLN(RUN,POLAR)
0120 IF(.NOT.DIAG) GO TO 17#
0121 WRITE(TERM,16#) RUN,POLAR,CDELECT,CDDATA
0122 16# FORMAT(418)
0123 17# CONTINUE
0124 RUN-I
0125 POLAR=MOD(POLAR+2,2)+1
0126 GO TO 2#
0127 18# CONTINUE
0128 RUN=RUN+1
0129 2# CONTINUE
0130 CALL CODLN(RUN,POLAR)
0131 IF(.NOT.DIAG) GO TO 21#
0132 WRITE(TERM,16#) RUN,POLAR,CDELECT,CDDATA
0133 GO TO 21#
0134 C
0135 C TWO-DIMENSIONAL CODING
0136 C
0137 68# CONTINUE
0138 STFBIT-STFBIT+1
0139 C
0140 C IF PREVIOUS LINE IS ONE-DIMENSIONAL, WRITE ONE EOL2
0141 C
0142 IF(MOD(INLNCT-2,K).EQ.8) CALL CODNG(11,8,8,8,8)
0143 C
0144 SET A8 TO LEFT EDGE-1 AND POLARITY-WHITE
0145 C
0146 A8=8
0147 POL=8
0148 LEFT=.TRUE.
0149 C
0150 C DETECT A1
0151 C
0152 62# CONTINUE
0153 I=A8+1
0154 IF(I.GT.PELMAX) GO TO 64#
0155 63# CONTINUE
0156 PEL=IAB(PELBUF(1,INCODE),1,1)
0157 IF(PEL.NE.POL) GO TO 64#
0158 I=I+1
0159 IF(I.LE.PELMAX) GO TO 63#
0160 64# CONTINUE
0161 A1=I
0162 C
0163 C DETECT B1
0164 C
0165 I=A8+1

```

PAGE 8884 ENCD2 5:13 PM SUN.. 31 OCT.. 1982

```

8166 IF(1.GT.PELMAX) GO TO 665
8167 IF(LEFT) GO TO 645
8168 PELM1-148(PELBUF(1,INREF),AB,1)
8169 GO TO 658
8170 645 PELM1-B
8171 658 CONTINUE
8172 PEL-148(PELBUF(1,INREF),1,1)
8173 IF(PEL.NE.PELM1) GO TO 678
8174 668 CONTINUE
8175 PELM1-PEL
8176 1-1+1
8177 IF(1.LE.PELMAX) GO TO 658
8178 665 CONTINUE
8179 81-1
8180 GO TO 718
8181 678 CONTINUE
8182 IF(PEL.NE.POL) GO TO 698
8183 GO TO 668
8184 698 CONTINUE
8185 81-1
8186 POL-PEL
8187
8188 C DETECT B2
8189 C
8190 C
8191 I-81+1
8192 IF(1.GT.PELMAX) GO TO 718
8193 788 CONTINUE
8194 PEL-148(PELBUF(1,INREF),1,1)
8195 IF(PEL.NE.POL) GO TO 728
8196 1-1+1
8197 IF(1.LE.PELMAX) GO TO 788
8198 718 CONTINUE
8199 82-1
8200 GO TO 738
8201 728 CONTINUE
8202 82-1
8203 POL-PEL
8204 738 CONTINUE
8205 IF(.NOT.LEFT) POLAR-148(PELBUF(1,INCD),AB,1)+1
8206 IF(.NOT.LEFT) GO TO 748
8207 POLAR-1
8208 AB-1
8209 LEFT-.FALSE.
8210 748 CONTINUE
8211 C
8212 C TEST FOR PASS MODE
8213 C
8214 IF(B2.GE.A1) GO TO 758
8215 C
8216 PASS MODE CODING (CAN'T END A LINE IN PASS MODE; NEW AB MUST HAVE
8217 SAME POLARITY AS B2)
8218 CALL CODNG(1,B,B,B,B)
8219 AB-B2
8220 GO TO 628

```

PAGE 8885 ENCD2 5:13 PM SUN., 31 OCT., 1982

```

8221 C 75# CONTINUE
8222 C
8223 MAB-IABS(A1-B1)
8224 IF(MAB-3) 835,835,799
8225 C
8226 C CODE BY HORIZONTAL MODE;FIRST DETECT A2
8227 C
8228 799 CONTINUE
8229 I-A1+1
8230 IF(I.GT.PELMAX) GO TO 81#
8231 C
8232 C CALCULATE POLARITY OF A1
8233 C
8234 POL-I48(PELBUF(1,INCD),A1,1)
8235 8# CONTINUE
8236 PEL-I48(PELBUF(1,INCD),1,1)
8237 IF(PEL.NE.POL) GO TO 82#
8238 I-1+1
8239 IF(1.LE.PELMAX) GO TO 8#
8240 81# A2-PELMAX+1
8241 GO TO 83#
8242 82# CONTINUE
8243 A2-I
8244 83# CONTINUE
8245 CALL CODNG(2,POLAR,A#A1,A2)
8246 A#-A2
8247 GO TO 96#
8248 C
8249 C CODE BY VERTICAL MODE
8250 C
8251 835 CONTINUE
8252 IF(A1-B1) 85#,84#,84#
8253 C
8254 84# CALL CODNG(A1-B1+3,B,B,B,B)
8255 GO TO 95#
8256 85# CONTINUE
8257 CALL CODNG(B1-A1+6,B,B,B,B)
8258 95# CONTINUE
8259 A#-A1
8260 C
8261 C TEST FOR END OF LINE
8262 C
8263 96# CONTINUE
8264 IF(A#-GT.PELMAX) GO TO 21#
8265 POL-I48(PELBUF(1,INCD),A#,1)
8266 GO TO 62#
8267 21# CONTINUE
8268 C
8269 C SWITCH CODE & REFERENCE LINES
8270 C
8271 TEMP-IMREF
8272 IMREF-INCD
8273 INCD-TEMP
8274 C
8275 COELV-(COELCT+NBPW-1)/NBPW

```

PAGE 0006 ENCD2 5:13 PM SUN., 31 OCT., 1982

```

0276      DO 300 I=2,CDELW
0277      STBUF(1)=CDBUF(1)
0278      300 CONTINUE
0279      C      SAVE LINE LENGTH(DATA BITS ONLY)
0280      C
0281      STAT(1)=CDDATA
0282      CALL WRITE(IDC89,IERR,STAT)
0283      IF(IERR.LT.0)STOP 300
0284      C
0285      C      COMPUTE STATISTICS
0286      C
0287      390 CONTINUE
0288      TCDEL=TCDEL+CDELCT-NBPW
0289      TCDATA=TCDATA+CDDATA
0290      IF(DIAG) WRITE(TERM,160) INLNCT, CDDATA
0291      C
0292      IF (.NOT.DIAG) GO TO 460
0293      CDELW=(CDELCT+NBPW-1)/NBPW
0294      WRITE(LPFIL,450) (CDBUF(I),I=1,CDELW)
0295      WRITE(LPFIL,450) (STBUF(I),I=1,CDELW)
0296      450 FORMAT(80I2)
0297      460 CONTINUE
0298      RETURN
0299      C
0300      C
0301      E N D

```

FTN4 COMPILER: HP92060-16092 REV. 2026 (800423)

-- NO WARNINGS -- NO ERRORS -- PROGRAM - 00842 COMMON - 00000

A-16

PAGE 0001 FTM. 5:14 PM SUN., 31 OCT., 1982

```

0001      FTM4.L.T.C
0002      SUBROUTINE GETL2(LBITS,MODE,VRD,L)
0003
0004      IMPLICIT INTEGER(A-Z)
0005
0006      LBITS - NO. OF BITS REQUESTED
0007      MODE -
0008          1 - NORMAL RETURN
0009          2 - EOL1 DETECTED
0010          3 - EOL2 DETECTED
0011          4 - NOT USED
0012      VRD - CONTAINS BITS RETURNED
0013      L - NO. OF BITS RETURNED
0014
0015      C*****LABELLED COMMON /G16BT/ *****
0016
0017      DIMENSION MASK(16),COMASK(16),LIBIT(16),LZBIT(16)
0018      COMMON /G16BT/MASK,COMASK,LIBIT,LZBIT,NBPW
0019
0020      C*****LABELLED COMMON ARRAYS*****
0021
0022      DIMENSION PELREF(258),PELCOD(258),OUTREF(258),OUTCOD(258)
0023      EQUIVALENCE (PELREF,PELBUF),(PELCOD,PELBUF(1,2))
0024      EQUIVALENCE (OUTREF,OTBUF),(OUTCOD,OTBUF(1,2))
0025      COMMON/BUFF/PELBUF(258,2),OTBUF(258,2)
0026      CDBUF(1024),STFBUF(1024),STAT(1)
0027      COMMON/HUFF/CODE(3,105,2),CODERD(3,11)
0028      COMMON/ERAY/ERRORS(100)
0029
0030      C*****LABELLED COMMON VARIABLES*****
0031
0032      COMMON/IVAR/PELMAX,VRES,EPHASE,CMPPHAX,ERRMOD,LINMAX,K
0033      COMMON/PVAR/INLNNO,OTLNNO,OTELV,CDELV,CDELP,CDDATA,
0034      CDELCT,INELCT,ICDATA,TCDEL,ERRPNT,ERRPNT,ERRLIM,
0035      ERRCNT,INLNCT,CONSEC,LNNOBF,ZERO,
0036      INCOD,INREF,OTCOD,OTREF,STFBIT,ERRCOR,BUFDIM
0037      COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0038      OUTF
0039      LOGICAL SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0040      OUTF
0041
0042      DOUBLE PRECISION TCDATA,TCDEL
0043
0044      C*****END COMMON*****
0045
0046      INTEGER VRD
0047
0048      C***** BEGIN PROGRAM *****
0049
0050      RETRIEVE NEXT BIT FROM CDBUF
0051
0052      ENCODE A NEW LINE IF NECESSARY
0053
0054      IF(LBITS+CDELP-1.LE.CDELCT) GO TO 200
0055

```

PAGE 0002 GETL2 5:14 PM SUN., 31 OCT., 1982

```

0056 IF(CDELCT-CDELP+1) 170,190,180
0057 170 STOP 170
0058 180 CONTINUE
0059 STFBUF(1)=148(STFBUF,CDELP,CDELCT-CDELP+1)
0060 190 CONTINUE
0061 CDELP=MBPV-(CDELCT-CDELP)
0062 CALL ENCD2
0063 200 CONTINUE
0064 VRD=148(STFBUF,CDELP,181TS)
0065 L=181TS
0066 IF(L.LT.13) GO TO 250
0067 IF(L.EQ.13.AND.VRD.EQ.CODERD(3,18)) GO TO 300
0068 IF(L.EQ.13.AND.VRD.EQ.CODERD(3,11)) GO TO 400
0069 250 CONTINUE
0070 MODE=1
0071 RETURN
0072 300 CONTINUE
0073 MODE=2
0074 RETURN
0075 400 CONTINUE
0076 MODE=3
0077 RETURN
0078 END

```

FTN4 COMPILER: HP92000-16092 REV. 2026 (800423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00126 COMMON = 00000

PAGE 0001 FTM. 5:14 PM SUN., 31 OCT., 1982

```

0001      FTM4,L,T,C
0002      BLOCK DATA MBLK2
0003      C
0004      IMPLICIT INTEGER(A-Z)
0005      C
0006      *****LABELLED COMMON ARRAYS*****
0007      C
0008      DIMENSION PELREF(258),PELCOD(258),OUTREF(258),OUTCOD(258)
0009      EQUIVALENCE (PELREF,PELBUF),(PELCOD,PELBUF(1,2))
0010      EQUIVALENCE (OUTREF,OTRBUF),(OUTCOD,OTRBUF(1,2))
0011      COMMON/BUFF/PELBUF(258,2),OTRBUF(258,2),STAT(1)
0012      & CDBUF(1824),STFBUF(1824),CODERD(3,11)
0013      COMMON/NUFF/CODE(3,185,2),CODERD(3,11)
0014      COMMON/ERAY/ERRORS(188)
0015      C
0016      *****FILE BUFFERS*****
0017      C
0018      COMMON/FILES/TERM,LPFIL,IDC87(144),IDC88(144),IDC89(144)
0019      INTEGER TERM,LPFIL,IDC87,IDC88,IDC89
0020      C
0021      IDC87 - PELFIL
0022      IDC88 - OTFIL
0023      IDC89 - STATFL
0024      C
0025      *****LABELLED COMMON VARIABLES*****
0026      C
0027      COMMON/IVAR/PELMAX,VRES,EPHASE,CMFMAX,ERRMOD,LINMAX,K
0028      COMMON/PVAR/INLNNO,OTLNNO,OTELV,CDELP,OTELP,CDELW,CDDATA,
0029      CDELCT,INELCT,TCDATA,TCDEL,ERRPNT,ERROFF,ERRLIM,
0030      ERRCNT,INLNCT,CONSEC,LNNOBF,ZERO
0031      INCOD,INREF,OTCOD,OTREF,STFBIT,ERRCOR,BUFDIM
0032      COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0033      OUTF
0034      LOGICAL SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0035      OUTF
0036      C
0037      DOUBLE PRECISION TCDATA,TCDEL
0038      C
0039      *****END COMMON*****
0040      C
0041      DATA TERM/1/
0042      DATA LPFIL/6/
0043      DATA DIAG/.FALSE./
0044      DATA WHITE/.FALSE./
0045      DATA SEARCH/.TRUE./
0046      DATA SYNC/.FALSE./
0047      DATA FILEND/.FALSE./
0048      DATA OUTF/.TRUE./
0049      DATA ERRCNT/8/
0050      DATA INREF/1/
0051      DATA INCOD/2/
0052      DATA OTREF/1/
0053      DATA OTCOD/2/
0054      DATA INLNNO/8/
0055      DATA INLNCT/8/

```

PAGE 0002 FTM. 5:14 PM SUN., 31 OCT., 1982

0056 DATA CONSEC/1/
0057 DATA STFLP/1/
0058 DATA STFLP/1/
0059 DATA TCDEL/5.000/
0060 DATA TCDEL/5.000/
C
0061 DATA CODE(1, 1,1),CODE(2, 1,1),CODE(3, 1,1)/ 8, 70, 558/
0062 DATA CODE(1, 2,1),CODE(2, 2,1),CODE(3, 2,1)/ 6, 90, 78/
0063 DATA CODE(1, 3,1),CODE(2, 3,1),CODE(3, 3,1)/ 4, 4, 78/
0064 DATA CODE(1, 4,1),CODE(2, 4,1),CODE(3, 4,1)/ 4, 5, 100/
0065 DATA CODE(1, 5,1),CODE(2, 5,1),CODE(3, 5,1)/ 4, 6, 138/
0066 DATA CODE(1, 6,1),CODE(2, 6,1),CODE(3, 6,1)/ 4, 7, 148/
0067 DATA CODE(1, 7,1),CODE(2, 7,1),CODE(3, 7,1)/ 4, 8, 168/
0068 DATA CODE(1, 8,1),CODE(2, 8,1),CODE(3, 8,1)/ 4, 9, 178/
0069 DATA CODE(1, 9,1),CODE(2, 9,1),CODE(3, 9,1)/ 5, 10, 238/
0070 DATA CODE(1, 10,1),CODE(2, 10,1),CODE(3, 10,1)/ 5, 11, 248/
0071 DATA CODE(1, 11,1),CODE(2, 11,1),CODE(3, 11,1)/ 5, 12, 78/
0072 DATA CODE(1, 12,1),CODE(2, 12,1),CODE(3, 12,1)/ 6, 65, 108/
0073 DATA CODE(1, 13,1),CODE(2, 13,1),CODE(3, 13,1)/ 6, 14, 108/
0074 DATA CODE(1, 14,1),CODE(2, 14,1),CODE(3, 14,1)/ 6, 15, 38/
0075 DATA CODE(1, 15,1),CODE(2, 15,1),CODE(3, 15,1)/ 6, 16, 648/
0076 DATA CODE(1, 16,1),CODE(2, 16,1),CODE(3, 16,1)/ 6, 17, 658/
0077 DATA CODE(1, 17,1),CODE(2, 17,1),CODE(3, 17,1)/ 6, 18, 528/
0078 DATA CODE(1, 18,1),CODE(2, 18,1),CODE(3, 18,1)/ 6, 19, 538/
0079 DATA CODE(1, 19,1),CODE(2, 19,1),CODE(3, 19,1)/ 7, 20, 478/
0080 DATA CODE(1, 20,1),CODE(2, 20,1),CODE(3, 20,1)/ 7, 21, 148/
0081 DATA CODE(1, 21,1),CODE(2, 21,1),CODE(3, 21,1)/ 7, 22, 108/
0082 DATA CODE(1, 22,1),CODE(2, 22,1),CODE(3, 22,1)/ 7, 23, 278/
0083 DATA CODE(1, 23,1),CODE(2, 23,1),CODE(3, 23,1)/ 7, 24, 38/
0084 DATA CODE(1, 24,1),CODE(2, 24,1),CODE(3, 24,1)/ 7, 25, 48/
0085 DATA CODE(1, 25,1),CODE(2, 25,1),CODE(3, 25,1)/ 7, 26, 508/
0086 DATA CODE(1, 26,1),CODE(2, 26,1),CODE(3, 26,1)/ 7, 27, 538/
0087 DATA CODE(1, 27,1),CODE(2, 27,1),CODE(3, 27,1)/ 7, 28, 238/
0088 DATA CODE(1, 28,1),CODE(2, 28,1),CODE(3, 28,1)/ 7, 29, 448/
0089 DATA CODE(1, 29,1),CODE(2, 29,1),CODE(3, 29,1)/ 7, 68, 308/
0090 DATA CODE(1, 30,1),CODE(2, 30,1),CODE(3, 30,1)/ 8, 31, 28/
0091 DATA CODE(1, 31,1),CODE(2, 31,1),CODE(3, 31,1)/ 8, 32, 38/
0092 DATA CODE(1, 32,1),CODE(2, 32,1),CODE(3, 32,1)/ 8, 33, 328/
0093 DATA CODE(1, 33,1),CODE(2, 33,1),CODE(3, 33,1)/ 8, 34, 338/
0094 DATA CODE(1, 34,1),CODE(2, 34,1),CODE(3, 34,1)/ 8, 35, 228/
0095 DATA CODE(1, 35,1),CODE(2, 35,1),CODE(3, 35,1)/ 8, 36, 238/
0096 DATA CODE(1, 36,1),CODE(2, 36,1),CODE(3, 36,1)/ 8, 37, 248/
0097 DATA CODE(1, 37,1),CODE(2, 37,1),CODE(3, 37,1)/ 8, 38, 258/
0098 DATA CODE(1, 38,1),CODE(2, 38,1),CODE(3, 38,1)/ 8, 39, 268/
0099 DATA CODE(1, 39,1),CODE(2, 39,1),CODE(3, 39,1)/ 8, 40, 278/
0100 DATA CODE(1, 40,1),CODE(2, 40,1),CODE(3, 40,1)/ 8, 41, 508/
0101 DATA CODE(1, 41,1),CODE(2, 41,1),CODE(3, 41,1)/ 8, 42, 618/
0102 DATA CODE(1, 42,1),CODE(2, 42,1),CODE(3, 42,1)/ 8, 43, 528/
0103 DATA CODE(1, 43,1),CODE(2, 43,1),CODE(3, 43,1)/ 8, 44, 538/
0104 DATA CODE(1, 44,1),CODE(2, 44,1),CODE(3, 44,1)/ 8, 45, 548/
0105 DATA CODE(1, 45,1),CODE(2, 45,1),CODE(3, 45,1)/ 8, 46, 558/
0106 DATA CODE(1, 46,1),CODE(2, 46,1),CODE(3, 46,1)/ 8, 47, 48/
0107 DATA CODE(1, 47,1),CODE(2, 47,1),CODE(3, 47,1)/ 8, 48, 58/
0108 DATA CODE(1, 48,1),CODE(2, 48,1),CODE(3, 48,1)/ 8, 49, 128/
0109 DATA CODE(1, 49,1),CODE(2, 49,1),CODE(3, 49,1)/ 8, 50, 138/
0110 DATA CODE(1, 50,1),CODE(2, 50,1),CODE(3, 50,1)/ 8, 51, 138/

A

PAGE 8885 FTN. 5:14 PM SUN., 31 OCT., 1982

8221 DATA CODE(1, 55,2),CODE(12, 55,2),CODE(13, 55,2)/12, 56, 788/
 8222 DATA CODE(1, 56,2),CODE(12, 56,2),CODE(13, 56,2)/12, 57, 478/
 8223 DATA CODE(1, 57,2),CODE(12, 57,2),CODE(13, 57,2)/12, 58, 588/
 8224 DATA CODE(1, 58,2),CODE(12, 58,2),CODE(13, 58,2)/12, 59, 1388/
 8225 DATA CODE(1, 59,2),CODE(12, 59,2),CODE(13, 59,2)/12, 60, 1388/
 8226 DATA CODE(1, 60,2),CODE(12, 60,2),CODE(13, 60,2)/12, 61, 538/
 8227 DATA CODE(1, 61,2),CODE(12, 61,2),CODE(13, 61,2)/12, 62, 548/
 8228 DATA CODE(1, 62,2),CODE(12, 62,2),CODE(13, 62,2)/12, 63, 1328/
 8229 DATA CODE(1, 63,2),CODE(12, 63,2),CODE(13, 63,2)/12, 64, 1468/
 8230 DATA CODE(1, 64,2),CODE(12, 64,2),CODE(13, 64,2)/12, 65, 1478/
 8231 DATA CODE(1, 65,2),CODE(12, 65,2),CODE(13, 65,2)/12, 66, 178/
 8232 DATA CODE(1, 66,2),CODE(12, 66,2),CODE(13, 66,2)/12, 67, 3188/
 8233 DATA CODE(1, 67,2),CODE(12, 67,2),CODE(13, 67,2)/12, 68, 3188/
 8234 DATA CODE(1, 68,2),CODE(12, 68,2),CODE(13, 68,2)/12, 69, 1338/
 8235 DATA CODE(1, 69,2),CODE(12, 69,2),CODE(13, 69,2)/12, 70, 638/
 8236 DATA CODE(1, 70,2),CODE(12, 70,2),CODE(13, 70,2)/12, 71, 648/
 8237 DATA CODE(1, 71,2),CODE(12, 71,2),CODE(13, 71,2)/12, 72, 1548/
 8238 DATA CODE(1, 72,2),CODE(12, 72,2),CODE(13, 72,2)/12, 73, 1548/
 8239 DATA CODE(1, 73,2),CODE(12, 73,2),CODE(13, 73,2)/12, 74, 1558/
 8240 DATA CODE(1, 74,2),CODE(12, 74,2),CODE(13, 74,2)/12, 75, 1138/
 8241 DATA CODE(1, 75,2),CODE(12, 75,2),CODE(13, 75,2)/12, 76, 1138/
 8242 DATA CODE(1, 76,2),CODE(12, 76,2),CODE(13, 76,2)/12, 77, 1148/
 8243 DATA CODE(1, 77,2),CODE(12, 77,2),CODE(13, 77,2)/12, 78, 1158/
 8244 DATA CODE(1, 78,2),CODE(12, 78,2),CODE(13, 78,2)/12, 79, 1628/
 8245 DATA CODE(1, 79,2),CODE(12, 79,2),CODE(13, 79,2)/12, 80, 1638/
 8246 DATA CODE(1, 80,2),CODE(12, 80,2),CODE(13, 80,2)/12, 81, 1648/
 8247 DATA CODE(1, 81,2),CODE(12, 81,2),CODE(13, 81,2)/12, 82, 1658/
 8248 DATA CODE(1, 82,2),CODE(12, 82,2),CODE(13, 82,2)/12, 83, 1658/
 8249 DATA CODE(1, 83,2),CODE(12, 83,2),CODE(13, 83,2)/12, 84, 1678/
 8250 DATA CODE(1, 84,2),CODE(12, 84,2),CODE(13, 84,2)/12, 85, 1228/
 8251 DATA CODE(1, 85,2),CODE(12, 85,2),CODE(13, 85,2)/12, 86, 1238/
 8252 DATA CODE(1, 86,2),CODE(12, 86,2),CODE(13, 86,2)/12, 87, 1248/
 8253 DATA CODE(1, 87,2),CODE(12, 87,2),CODE(13, 87,2)/12, 88, 1258/
 8254 DATA CODE(1, 88,2),CODE(12, 88,2),CODE(13, 88,2)/12, 89, 1328/
 8255 DATA CODE(1, 89,2),CODE(12, 89,2),CODE(13, 89,2)/12, 90, 1338/
 8256 DATA CODE(1, 90,2),CODE(12, 90,2),CODE(13, 90,2)/12, 91, 1448/
 8257 DATA CODE(1, 91,2),CODE(12, 91,2),CODE(13, 91,2)/12, 92, 1458/
 8258 DATA CODE(1, 92,2),CODE(12, 92,2),CODE(13, 92,2)/12, 93, 148/
 8259 DATA CODE(1, 93,2),CODE(12, 93,2),CODE(13, 93,2)/12, 94, 148/
 8260 DATA CODE(1, 94,2),CODE(12, 94,2),CODE(13, 94,2)/12, 95, 158/
 8261 DATA CODE(1, 95,2),CODE(12, 95,2),CODE(13, 95,2)/12, 96, 228/
 8262 DATA CODE(1, 96,2),CODE(12, 96,2),CODE(13, 96,2)/12, 97, 238/
 8263 DATA CODE(1, 97,2),CODE(12, 97,2),CODE(13, 97,2)/12, 98, 248/
 8264 DATA CODE(1, 98,2),CODE(12, 98,2),CODE(13, 98,2)/12, 99, 258/
 8265 DATA CODE(1, 99,2),CODE(12, 99,2),CODE(13, 99,2)/12, 100, 268/
 8266 DATA CODE(1, 100,2),CODE(12, 100,2),CODE(13, 100,2)/12, 101, 278/
 8267 DATA CODE(1, 101,2),CODE(12, 101,2),CODE(13, 101,2)/12, 102, 348/
 8268 DATA CODE(1, 102,2),CODE(12, 102,2),CODE(13, 102,2)/12, 103, 358/
 8269 DATA CODE(1, 103,2),CODE(12, 103,2),CODE(13, 103,2)/12, 104, 368/
 8270 DATA CODE(1, 104,2),CODE(12, 104,2),CODE(13, 104,2)/12, 105, 378/
 8271 DATA CODE(1, 105,2),CODE(12, 105,2),CODE(13, 105,2)/12, 106, 38/
 8272 DATA CODERD(1,1),CODERD(12,1),CODERD(13,1)/ 4, 5, 18/
 8273 DATA CODERD(1,2),CODERD(12,2),CODERD(13,2)/ 3, 4, 18/
 8274 DATA CODERD(1,3),CODERD(12,3),CODERD(13,3)/ 1, 2, 18/
 8275 DATA CODERD(1,4),CODERD(12,4),CODERD(13,4)/ 3, 7, 38/

PAGE 0006 FTM. 5.14 PM SUN., 31 OCT., 1982

```

0276 DATA CODERD(1,5),CODERD(2,5),CODERD(3,5)//
0277 DATA CODERD(1,6),CODERD(2,6),CODERD(3,6)//
0278 DATA CODERD(1,7),CODERD(2,7),CODERD(3,7)//
0279 DATA CODERD(1,8),CODERD(2,8),CODERD(3,8)//
0280 DATA CODERD(1,9),CODERD(2,9),CODERD(3,9)//
0281 DATA CODERD(1,10),CODERD(2,10),CODERD(3,10)//
0282 DATA CODERD(1,11),CODERD(2,11),CODERD(3,11)//
0283 C
0284 E N D

```

FTN4 COMPILER: HP92060-16092 REV. 2026 (800423)

** NO WARNINGS ** NO ERRORS **

```

BLOCK COMMON BUFF SIZE = 03001
BLOCK COMMON HUFF SIZE = 00663
BLOCK COMMON ERAY SIZE = 00100
BLOCK COMMON FILES SIZE = 00434
BLOCK COMMON IVAR SIZE = 00007
BLOCK COMMON PVAR SIZE = 00030
BLOCK COMMON LOGIC SIZE = 00010

```

PAGE 0001 FTM. 5:15 PM SUN., 31 OCT., 1982

```

0001      FTN4,L,T,C
0002      SUBROUTINE ONED2(INDEX,COLOR,STATUS,L)
0003      C
0004      IMPLICIT INTEGER(A-Z)
0005      C
0006      *****LABELLED COMMON ARRAYS*****
0007      C
0008      DIMENSION PELREF(258),PELCOD(258),OUTREF(258),OUTCOD(258)
0009      EQUIVALENCE (PELREF,PELBUF),(PELCOD,PELBUF(1,2))
0010      EQUIVALENCE (OUTREF,OTBUF),(OUTCOD,OTBUF(1,2))
0011      COMMON/BUF/PELBUF(258,2),OTBUF(258,2),STAT(1)
0012      & CDBUF(1024),STFBUF(1024),
0013      COMMON/HUFF/CODE(3,105,2),CODERD(3,11)
0014      COMMON/ERAY/ERRORS(100)
0015      C
0016      *****FILE BUFFERS*****
0017      C
0018      COMMON/FILES/TERM,LPFIL,IDCB7(144),IDCB8(144),IDCB9(144)
0019      INTEGER TERM,LPFIL,IDCB7,IDCB8,IDCB9
0020      C
0021      IDC87 - PELFIL
0022      IDC88 - OTFIL
0023      IDC89 - STATFL
0024      C
0025      *****LABELLED COMMON VARIABLES*****
0026      C
0027      COMMON/IVAR/PELMAX,VRES,EPHASE,CHPMAX,ERRMOD,LINMAX,K
0028      COMMON/PVAR/INLNNO,OTLNNO,OTELV,CDELP,OTELP,CDELV,CDDATA,
0029      CDELCI,INLNCT,TCDATA,TCDEL,ERRPNT,ERROFF,ERRLIM,
0030      ERRCNT,INLNCT,CONSEC,LNNOBF,ZERO,
0031      INCOD,INREF,OTCOD,OTREF,STFBIT,ERRCOR,BUFDIM
0032      COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0033      OUTF
0034      LOGICAL SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0035      OUTF
0036      C
0037      DOUBLE PRECISION TCDATA,TCDEL
0038      C
0039      *****END COMMON*****
0040      C
0041      INTEGER COLOR,I,INDEX,L,LBITS,LENBIT,MODE,RUNLEN,STATUS
0042      C
0043      *****BEGIN PROGRAM*****
0044      C
0045      C BEGIN DECODE LOOP; RETRIEVE NEXT CODE WORD LENGTH (L)
0046      C
0047      1000 CONTINUE
0048      LENBIT=CODE(1,INDEX,COLOR)
0049      CALL GETL2(LENBIT,MODE,LBITS,L)
0050      IF(DIAG) WRITE(TERM,1003) LENBIT,MODE,LBITS,L
0051      1003 FORMAT(2I6,08,16)
0052      GO TO (1040,1200,1205,1100), MODE
0053      1040 CONTINUE
0054      IF(LBITS.EQ.CODE(3,INDEX,COLOR)) GO TO 1100
0055      C

```

PAGE 0002 0NED2 5:15 PM SUN., 31 OCT., 1982

```

0056 C NO MATCH: ADVANCE CODE WORD INDEX VIA DECODE THREAD
0057 C
0058 INDEX-CODE(2,INDEX,COLOR)
0059 IF(INDEX-GE.100) GO TO 1190
0060 IF(CODE(1,INDEX,COLOR).EQ.LENBIT) GO TO 1040
0061 C
0062 C CODE WORD LONGER: FROM THE TOP
0063 C
0064 C GO TO 1002
0065 C
0066 C MATCH FOUND
0067 C
0068 C 1100 CONTINUE
0069 C CDELP=CDELP+L
0070 C
0071 C NOT AN EOL
0072 C
0073 C
0074 C
0075 C
0076 C
0077 C
0078 C
0079 C
0080 C
0081 C
0082 C
0083 C
0084 C
0085 C
0086 C
0087 C
0088 C
0089 C
0090 C
0091 C
0092 C
0093 C
0094 C
0095 C
0096 C
0097 C
0098 C
0099 C
0100 C
0101 C
0102 C
0103 C
0104 C
0105 C
0106 C
0107 C
0108 C
0109 C
0110 C

```

TEST FOR MAKE UP OR TERMINATING CODE

```

0081 C RUNLEN=INDEX-1
0082 C IF(INDEX-GE.65) RUNLEN=(INDEX-64)*64
0083 C IF(RUNLEN.EQ.0) GO TO 1160
0084 C IF(COLOR.EQ.1) GO TO 1155
0085 C IF(RUNLEN.LT.0) STOP 1100
0086 C
0087 C
0088 C
0089 C
0090 C
0091 C
0092 C
0093 C
0094 C
0095 C
0096 C
0097 C
0098 C
0099 C
0100 C
0101 C
0102 C
0103 C
0104 C
0105 C
0106 C
0107 C
0108 C
0109 C
0110 C

```

ADD BLACK RUN TO OUTPUT BUFFER

```

0081 C DO 1150 I=1,RUNLEN
0082 C CALL M12(COLOR-1,OTBUF(1,OTCOD),OTELP,1)
0083 C OTELP=OTELP+1
0084 C IF(OTELP-1.GT.PELMAX) GO TO 1100
0085 C
0086 C
0087 C
0088 C
0089 C
0090 C
0091 C
0092 C
0093 C
0094 C
0095 C
0096 C
0097 C
0098 C
0099 C
0100 C
0101 C
0102 C
0103 C
0104 C
0105 C
0106 C
0107 C
0108 C
0109 C
0110 C

```

ADD WHITE RUN TO OUTPUT BUFFER (BY DEFAULT)

```

0081 C DO 1150 I=1,RUNLEN
0082 C CALL M12(COLOR-1,OTBUF(1,OTCOD),OTELP,1)
0083 C OTELP=OTELP+1
0084 C IF(OTELP-1.GT.PELMAX) GO TO 1100
0085 C
0086 C
0087 C
0088 C
0089 C
0090 C
0091 C
0092 C
0093 C
0094 C
0095 C
0096 C
0097 C
0098 C
0099 C
0100 C
0101 C
0102 C
0103 C
0104 C
0105 C
0106 C
0107 C
0108 C
0109 C
0110 C

```

OUTPUT LINE LESS THAN OR EQUAL TO MAX SPECIFIED

```

0081 C DO 1150 I=1,RUNLEN
0082 C CALL M12(COLOR-1,OTBUF(1,OTCOD),OTELP,1)
0083 C OTELP=OTELP+1
0084 C IF(OTELP-1.GT.PELMAX) GO TO 1100
0085 C
0086 C
0087 C
0088 C
0089 C
0090 C
0091 C
0092 C
0093 C
0094 C
0095 C
0096 C
0097 C
0098 C
0099 C
0100 C
0101 C
0102 C
0103 C
0104 C
0105 C
0106 C
0107 C
0108 C
0109 C
0110 C

```

CONTINUE

```

0081 C DO 1150 I=1,RUNLEN
0082 C CALL M12(COLOR-1,OTBUF(1,OTCOD),OTELP,1)
0083 C OTELP=OTELP+1
0084 C IF(OTELP-1.GT.PELMAX) GO TO 1100
0085 C
0086 C
0087 C
0088 C
0089 C
0090 C
0091 C
0092 C
0093 C
0094 C
0095 C
0096 C
0097 C
0098 C
0099 C
0100 C
0101 C
0102 C
0103 C
0104 C
0105 C
0106 C
0107 C
0108 C
0109 C
0110 C

```

RUN ADDED TO OUTPUT LINE; LENGTH LESS THAN OR EQUAL TO PELMAX (1)

```

0081 C DO 1150 I=1,RUNLEN
0082 C CALL M12(COLOR-1,OTBUF(1,OTCOD),OTELP,1)
0083 C OTELP=OTELP+1
0084 C IF(OTELP-1.GT.PELMAX) GO TO 1100
0085 C
0086 C
0087 C
0088 C
0089 C
0090 C
0091 C
0092 C
0093 C
0094 C
0095 C
0096 C
0097 C
0098 C
0099 C
0100 C
0101 C
0102 C
0103 C
0104 C
0105 C
0106 C
0107 C
0108 C
0109 C
0110 C

```

CONTINUE

```

0081 C DO 1150 I=1,RUNLEN
0082 C CALL M12(COLOR-1,OTBUF(1,OTCOD),OTELP,1)
0083 C OTELP=OTELP+1
0084 C IF(OTELP-1.GT.PELMAX) GO TO 1100
0085 C
0086 C
0087 C
0088 C
0089 C
0090 C
0091 C
0092 C
0093 C
0094 C
0095 C
0096 C
0097 C
0098 C
0099 C
0100 C
0101 C
0102 C
0103 C
0104 C
0105 C
0106 C
0107 C
0108 C
0109 C
0110 C

```

CHCOL=.TRUE.
STATUS=1
RETURN

PAGE 0003 ONED2 5:15 PM SUN., 31 OCT., 1982

```

0111 C RUN ADDED UNTIL PELMAX EXCEEDED; LINE TOO LONG (2)
0112 C
0113 1180 CONTINUE
0114 IF(DIAG) WRITE(TERM,1185) (OTBUF(I,OTCOD),I-1,BUFDIM)
0115 1185 FORMAT(1508)
0116 STATUS=2
0117 RETURN
0118 C
0119 C NO MATCH FOUND IN CODE TABLE (3)
0120 C
0121 1190 CONTINUE
0122 STATUS=3
0123 RETURN
0124 C
0125 C EOL1 DETECTED (4)
0126 C
0127 1200 CONTINUE
0128 STATUS=4
0129 RETURN
0130 C
0131 C EOL2 DETECTED (5)
0132 C
0133 1205 CONTINUE
0134 STATUS=5
0135 RETURN
0136 END

```

FTNA COMPILER: HP92060-16092 REV. 2026 (000423)

** NO WARNINGS ** NO ERRORS ** PROGRAM - 00271 COMMON - 00000

```

0001      FTM4,L,T,C
0002      SUBROUTINE TV002( INDEX,COLOR,STATUS,L)
0003
0004      IMPLICIT INTEGER(A-Z)
0005
0006      C*****LABELLED COMMON ARRAYS*****
0007      C
0008      DIMENSION PELREF(258),PELCOD(258),OUTREF(258),OUTCOD(258)
0009      EQUIVALENCE (PELREF,PELBUF),(PELCOD,PELBUF(1,2))
0010      EQUIVALENCE (OUTREF,OTBUF),(OUTCOD,OTBUF(1,2))
0011      COMMON/BUFF/PELBUF(258,2),OTBUF(258,2)
0012      COMMON/HUFF/CODE(3,125,2),CODERD(3,11)
0013      COMMON/ERAY/ERRORS(100)
0014
0015      C*****FILE BUFFERS*****
0016      C
0017      COMMON/FILES/TERM,LPFIL,IDCB7(144),IDCB8(144),IDCB9(144)
0018      INTEGER TERM,LPFIL,IDCB7,IDCB8,IDCB9
0019
0020      C
0021      IDC87 - PELFIL
0022      IDC88 - OTFIL
0023      IDC89 - STATFL
0024
0025      C*****LABELLED COMMON VARIABLES*****
0026      C
0027      COMMON/IVAR/PELMAX,VRES,EPHASE,CMPHAX,ERRMOD,LINMAX,K
0028      COMMON/PVAR/INLNO,OTLNO,OTELV,CDELP,OTELP,CDELV,CDDATA,
0029      CDELCCT,INELCT,TCDATA,TCDEL,ERRPNT,ERROFF,ERRLIM,
0030      ERRCNT,INLNCT,CONSEC,LNNOBF,ZERO,
0031      INCOD,INREF,OTCOD,OTREF,STFBIT,ERRCOR,BUFDIM
0032      COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0033      OUTF
0034      LOGICAL SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0035      OUTF
0036
0037      C
0038      DOUBLE PRECISION TCDATA,TCDEL
0039
0040      C*****END COMMON*****
0041      C
0042      INTEGER AB,B1,B2,COLOR,ENTRY,I,INDEX,L,LBITS,LENBIT,MODE
0043      INTEGER PEL,PELMI,POL,RUNLEN,STATE,STATUS
0044
0045      C*****BEGIN PROGRAM*****
0046      C
0047      C
0048      C
0049      C
0050      C
0051      C
0052      C
0053      C
0054      C
0055      C
0056      C
0057      C
0058      C
0059      C
0060      C
0061      C
0062      C
0063      C
0064      C
0065      C
0066      C
0067      C
0068      C
0069      C
0070      C
0071      C
0072      C
0073      C
0074      C
0075      C
0076      C
0077      C
0078      C
0079      C
0080      C
0081      C
0082      C
0083      C
0084      C
0085      C
0086      C
0087      C
0088      C
0089      C
0090      C
0091      C
0092      C
0093      C
0094      C
0095      C
0096      C
0097      C
0098      C
0099      C
0100      C
0101      C
0102      C
0103      C
0104      C
0105      C
0106      C
0107      C
0108      C
0109      C
0110      C
0111      C
0112      C
0113      C
0114      C
0115      C
0116      C
0117      C
0118      C
0119      C
0120      C
0121      C
0122      C
0123      C
0124      C
0125      C
0126      C
0127      C
0128      C
0129      C
0130      C
0131      C
0132      C
0133      C
0134      C
0135      C
0136      C
0137      C
0138      C
0139      C
0140      C
0141      C
0142      C
0143      C
0144      C
0145      C
0146      C
0147      C
0148      C
0149      C
0150      C
0151      C
0152      C
0153      C
0154      C
0155      C
0156      C
0157      C
0158      C
0159      C
0160      C
0161      C
0162      C
0163      C
0164      C
0165      C
0166      C
0167      C
0168      C
0169      C
0170      C
0171      C
0172      C
0173      C
0174      C
0175      C
0176      C
0177      C
0178      C
0179      C
0180      C
0181      C
0182      C
0183      C
0184      C
0185      C
0186      C
0187      C
0188      C
0189      C
0190      C
0191      C
0192      C
0193      C
0194      C
0195      C
0196      C
0197      C
0198      C
0199      C
0200      C
0201      C
0202      C
0203      C
0204      C
0205      C
0206      C
0207      C
0208      C
0209      C
0210      C
0211      C
0212      C
0213      C
0214      C
0215      C
0216      C
0217      C
0218      C
0219      C
0220      C
0221      C
0222      C
0223      C
0224      C
0225      C
0226      C
0227      C
0228      C
0229      C
0230      C
0231      C
0232      C
0233      C
0234      C
0235      C
0236      C
0237      C
0238      C
0239      C
0240      C
0241      C
0242      C
0243      C
0244      C
0245      C
0246      C
0247      C
0248      C
0249      C
0250      C
0251      C
0252      C
0253      C
0254      C
0255      C
0256      C
0257      C
0258      C
0259      C
0260      C
0261      C
0262      C
0263      C
0264      C
0265      C
0266      C
0267      C
0268      C
0269      C
0270      C
0271      C
0272      C
0273      C
0274      C
0275      C
0276      C
0277      C
0278      C
0279      C
0280      C
0281      C
0282      C
0283      C
0284      C
0285      C
0286      C
0287      C
0288      C
0289      C
0290      C
0291      C
0292      C
0293      C
0294      C
0295      C
0296      C
0297      C
0298      C
0299      C
0300      C
0301      C
0302      C
0303      C
0304      C
0305      C
0306      C
0307      C
0308      C
0309      C
0310      C
0311      C
0312      C
0313      C
0314      C
0315      C
0316      C
0317      C
0318      C
0319      C
0320      C
0321      C
0322      C
0323      C
0324      C
0325      C
0326      C
0327      C
0328      C
0329      C
0330      C
0331      C
0332      C
0333      C
0334      C
0335      C
0336      C
0337      C
0338      C
0339      C
0340      C
0341      C
0342      C
0343      C
0344      C
0345      C
0346      C
0347      C
0348      C
0349      C
0350      C
0351      C
0352      C
0353      C
0354      C
0355      C
0356      C
0357      C
0358      C
0359      C
0360      C
0361      C
0362      C
0363      C
0364      C
0365      C
0366      C
0367      C
0368      C
0369      C
0370      C
0371      C
0372      C
0373      C
0374      C
0375      C
0376      C
0377      C
0378      C
0379      C
0380      C
0381      C
0382      C
0383      C
0384      C
0385      C
0386      C
0387      C
0388      C
0389      C
0390      C
0391      C
0392      C
0393      C
0394      C
0395      C
0396      C
0397      C
0398      C
0399      C
0400      C
0401      C
0402      C
0403      C
0404      C
0405      C
0406      C
0407      C
0408      C
0409      C
0410      C
0411      C
0412      C
0413      C
0414      C
0415      C
0416      C
0417      C
0418      C
0419      C
0420      C
0421      C
0422      C
0423      C
0424      C
0425      C
0426      C
0427      C
0428      C
0429      C
0430      C
0431      C
0432      C
0433      C
0434      C
0435      C
0436      C
0437      C
0438      C
0439      C
0440      C
0441      C
0442      C
0443      C
0444      C
0445      C
0446      C
0447      C
0448      C
0449      C
0450      C
0451      C
0452      C
0453      C
0454      C
0455      C
0456      C
0457      C
0458      C
0459      C
0460      C
0461      C
0462      C
0463      C
0464      C
0465      C
0466      C
0467      C
0468      C
0469      C
0470      C
0471      C
0472      C
0473      C
0474      C
0475      C
0476      C
0477      C
0478      C
0479      C
0480      C
0481      C
0482      C
0483      C
0484      C
0485      C
0486      C
0487      C
0488      C
0489      C
0490      C
0491      C
0492      C
0493      C
0494      C
0495      C
0496      C
0497      C
0498      C
0499      C
0500      C
0501      C
0502      C
0503      C
0504      C
0505      C
0506      C
0507      C
0508      C
0509      C
0510      C
0511      C
0512      C
0513      C
0514      C
0515      C
0516      C
0517      C
0518      C
0519      C
0520      C
0521      C
0522      C
0523      C
0524      C
0525      C
0526      C
0527      C
0528      C
0529      C
0530      C
0531      C
0532      C
0533      C
0534      C
0535      C
0536      C
0537      C
0538      C
0539      C
0540      C
0541      C
0542      C
0543      C
0544      C
0545      C
0546      C
0547      C
0548      C
0549      C
0550      C
0551      C
0552      C
0553      C
0554      C
0555      C
0556      C
0557      C
0558      C
0559      C
0560      C
0561      C
0562      C
0563      C
0564      C
0565      C
0566      C
0567      C
0568      C
0569      C
0570      C
0571      C
0572      C
0573      C
0574      C
0575      C
0576      C
0577      C
0578      C
0579      C
0580      C
0581      C
0582      C
0583      C
0584      C
0585      C
0586      C
0587      C
0588      C
0589      C
0590      C
0591      C
0592      C
0593      C
0594      C
0595      C
0596      C
0597      C
0598      C
0599      C
0600      C
0601      C
0602      C
0603      C
0604      C
0605      C
0606      C
0607      C
0608      C
0609      C
0610      C
0611      C
0612      C
0613      C
0614      C
0615      C
0616      C
0617      C
0618      C
0619      C
0620      C
0621      C
0622      C
0623      C
0624      C
0625      C
0626      C
0627      C
0628      C
0629      C
0630      C
0631      C
0632      C
0633      C
0634      C
0635      C
0636      C
0637      C
0638      C
0639      C
0640      C
0641      C
0642      C
0643      C
0644      C
0645      C
0646      C
0647      C
0648      C
0649      C
0650      C
0651      C
0652      C
0653      C
0654      C
0655      C
0656      C
0657      C
0658      C
0659      C
0660      C
0661      C
0662      C
0663      C
0664      C
0665      C
0666      C
0667      C
0668      C
0669      C
0670      C
0671      C
0672      C
0673      C
0674      C
0675      C
0676      C
0677      C
0678      C
0679      C
0680      C
0681      C
0682      C
0683      C
0684      C
0685      C
0686      C
0687      C
0688      C
0689      C
0690      C
0691      C
0692      C
0693      C
0694      C
0695      C
0696      C
0697      C
0698      C
0699      C
0700      C
0701      C
0702      C
0703      C
0704      C
0705      C
0706      C
0707      C
0708      C
0709      C
0710      C
0711      C
0712      C
0713      C
0714      C
0715      C
0716      C
0717      C
0718      C
0719      C
0720      C
0721      C
0722      C
0723      C
0724      C
0725      C
0726      C
0727      C
0728      C
0729      C
0730      C
0731      C
0732      C
0733      C
0734      C
0735      C
0736      C
0737      C
0738      C
0739      C
0740      C
0741      C
0742      C
0743      C
0744      C
0745      C
0746      C
0747      C
0748      C
0749      C
0750      C
0751      C
0752      C
0753      C
0754      C
0755      C
0756      C
0757      C
0758      C
0759      C
0760      C
0761      C
0762      C
0763      C
0764      C
0765      C
0766      C
0767      C
0768      C
0769      C
0770      C
0771      C
0772      C
0773      C
0774      C
0775      C
0776      C
0777      C
0778      C
0779      C
0780      C
0781      C
0782      C
0783      C
0784      C
0785      C
0786      C
0787      C
0788      C
0789      C
0790      C
0791      C
0792      C
0793      C
0794      C
0795      C
0796      C
0797      C
0798      C
0799      C
0800      C
0801      C
0802      C
0803      C
0804      C
0805      C
0806      C
0807      C
0808      C
0809      C
0810      C
0811      C
0812      C
0813      C
0814      C
0815      C
0816      C
0817      C
0818      C
0819      C
0820      C
0821      C
0822      C
0823      C
0824      C
0825      C
0826      C
0827      C
0828      C
0829      C
0830      C
0831      C
0832      C
0833      C
0834      C
0835      C
0836      C
0837      C
0838      C
0839      C
0840      C
0841      C
0842      C
0843      C
0844      C
0845      C
0846      C
0847      C
0848      C
0849      C
0850      C
0851      C
0852      C
0853      C
0854      C
0855      C
0856      C
0857      C
0858      C
0859      C
0860      C
0861      C
0862      C
0863      C
0864      C
0865      C
0866      C
0867      C
0868      C
0869      C
0870      C
0871      C
0872      C
0873      C
0874      C
0875      C
0876      C
0877      C
0878      C
0879      C
0880      C
0881      C
0882      C
0883      C
0884      C
0885      C
0886      C
0887      C
0888      C
0889      C
0890      C
0891      C
0892      C
0893      C
0894      C
0895      C
0896      C
0897      C
0898      C
0899      C
0900      C
0901      C
0902      C
0903      C
0904      C
0905      C
0906      C
0907      C
0908      C
0909      C
0910      C
0911      C
0912      C
0913      C
0914      C
0915      C
0916      C
0917      C
0918      C
0919      C
0920      C
0921      C
0922      C
0923      C
0924      C
0925      C
0926      C
0927      C
0928      C
0929      C
0930      C
0931      C
0932      C
0933      C
0934      C
0935      C
0936      C
0937      C
0938      C
0939      C
0940      C
0941      C
0942      C
0943      C
0944      C
0945      C
0946      C
0947      C
0948      C
0949      C
0950      C
0951      C
0952      C
0953      C
0954      C
0955      C
0956      C
0957      C
0958      C
0959      C
0960      C
0961      C
0962      C
0963      C
0964      C
0965      C
0966      C
0967      C
0968      C
0969      C
0970      C
0971      C
0972      C
0973      C
0974      C
0975      C
0976      C
0977      C
0978      C
0979      C
0980      C
0981      C
0982      C
0983      C
0984      C
0985      C
0986      C
0987      C
0988      C
0989      C
0990      C
0991      C
0992      C
0993      C
0994      C
0995      C
0996      C
0997      C
0998      C
0999      C
1000      C

```

PAGE 0002 TV002 5:16 PM SUN., 31 OCT., 1982

```

0056 C NO MATCH; ADVANCE CODE WORD INDEX VIA DECODE THREAD
0057 C
0058 C INDEX-CODERD(2,INDEX)
0059 IF(INDEX.GE.12) GO TO 119B
0060 IF(CODERD(1,INDEX).EQ.LENBIT) GO TO 104B
0061 C
0062 C CODE WORD LONGER; FROM THE TOP
0063 C
0064 C GO TO 1052
0065 C
0066 C MATCH FOUND
0067 C
0068 C 119B CONTINUE
0069 C CDELP-CDELP+L
0070 C
0071 C NOT AN EOL
0072 C
0073 C FIND B1 AND B2
0074 C
0075 C
0076 C
0077 C AS-OTELP
0078 IF(OTELP.EQ.1) AS-B
0079 POL-COLOR-1
0080 C
0081 C DETECT B1
0082 C
0083 C 1-AS+1
0084 IF(1.GT.PELMAX) GO TO 65
0085 PELM1-B
0086 IF(AS.EQ.B) GO TO 5B
0087 PELM1-14B(OTBUF(1,OTREF),AS,1)
0088 CONTINUE
0089 PEL-14B(OTBUF(1,OTREF),1,1)
0090 IF(PEL.WE.PELM1) GO TO 7B
0091 C
0092 C CONTINUE
0093 PELM1-PEL
0094 1-1+1
0095 IF(1.LE.PELMAX) GO TO 5B
0096 B1-1
0097 GO TO 92
0098 CONTINUE
0099 IF(PEL.WE.POL) GO TO 9B
0100 GO TO 6B
0101 CONTINUE
0102 B1-1
0103 POL-PEL
0104 C
0105 C DETECT B2
0106 C
0107 C 1-B1+1
0108 IF(1.GT.PELMAX) GO TO 92
0109 CONTINUE
0110 PEL-14B(OTBUF(1,OTREF),1,1)

```

PAGE 8083 TV002 5:16 PM SUN., 31 OCT., 1982

```

8111 IF(PEL.NE.POL) GO TO 92
8112 I=I+1
8113 IF(I.LE.PELMAX) GO TO 91
8114 92 CONTINUE
8115 B2=I
8116 GO TO (108,208,308,408,408,408,608,608,608),INDEX
8117 C
8118 C PASS MODE
8119 C
8120 108 CONTINUE
8121 RUNLEN=B2-OTELP
8122 CHCOL=.FALSE.
8123 GO TO (1155,1145),COLOR
8124 C
8125 C HORIZONTAL MODE
8126 C
8127 208 CONTINUE
8128 ENTRY=3
8129 CALL OMED2(ENTRY,COLOR,STATE,L)
8130 GO TO (218,1188,1198,1208,1208),STATE
8131 218 CONTINUE
8132 COLOR=MOD(COLOR+2,2)+1
8133 ENTRY=3
8134 CALL OMED2(ENTRY,COLOR,STATE,L)
8135 GO TO (228,1188,1198,1208,1208),STATE
8136 228 CONTINUE
8137 CHCOL=.TRUE.
8138 GO TO 1168
8139 C
8140 C VERTICAL MODE AIB1=B
8141 C
8142 308 CONTINUE
8143 RUNLEN=B1-OTELP
8144 CHCOL=.TRUE.
8145 GO TO (1155,1145),COLOR
8146 C
8147 C VERTICAL MODE VR1 AIB1=1,2,3
8148 C
8149 408 CONTINUE
8150 RUNLEN=B1-OTELP+INDEX-3
8151 CHCOL=.TRUE.
8152 GO TO (1155,1145),COLOR
8153 C
8154 C
8155 C VERTICAL MODE LEFT VLI AIB1=1,2,3
8156 C
8157 608 CONTINUE
8158 RUNLEN=B1-OTELP-(INDEX-6)
8159 CHCOL=.TRUE.
8160 GO TO (1155,1145),COLOR
8161 C
8162 C ADD BLACK RUN TO OUTPUT BUFFER
8163 C
8164 1145 CONTINUE
8165 IF(RUNLEN) 1198,1168,1147

```

PAGE 8884 TV002 .16 PM SUN., 31 OCT., 1982

```

8166 1147 CONTINUE
8167 DO 1150 I=1,RUNLEN
8168 CALL MI28(COLOR-1,OTBUF(1,OTCOD),OTELP,I)
8169 OTELP=OTELP+1
8170 IF(OTELP-1.GT.PELMAX) GO TO 1180
8171 1150 CONTINUE
8172 GO TO 1160
8173 C
8174 C ADD WHITE RUN TO OUTPUT BUFFER (BY DEFAULT)
8175 C
8176 1155 CONTINUE
8177 IF(RUNLEN.LT.8) GO TO 1190
8178 OTELP=OTELP+RUNLEN
8179 IF(OTELP-1.GT.PELMAX) GO TO 1180
8180 C
8181 C RUN ADDED TO OUTPUT LINE; LENGTH LESS THAN OR EQUAL TO PELMAX (1)
8182 C
8183 1160 CONTINUE
8184 STATUS=1
8185 RETURN
8186 C
8187 C RUN ADDED UNTIL PELMAX EXCEEDED; LINE TOO LONG (2)
8188 C
8189 1180 CONTINUE
8190 IF(DIAG) WRITE(TERM,1185) (OTBUF(1,OTCOD),I=1,BUFDIM)
8191 1105 FORMAT(1508)
8192 STATUS=2
8193 RETURN
8194 C
8195 C NO MATCH FOUND IN CODE TABLE (3)
8196 C
8197 1190 CONTINUE
8198 STATUS=3
8199 RETURN
8200 C
8201 C EOL1 DETECTED (4)
8202 C
8203 1200 CONTINUE
8204 STATUS=4
8205 RETURN
8206 C
8207 C EOL2 DETECTED (5)
8208 C
8209 1205 CONTINUE
8210 STATUS=5
8211 RETURN
8212 END

```

FTN4 COMPILER: HP92060-16892 REV. 2026 (888423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 88476 COMMON = 88888

PAGE 0001 FTN. 5:03 PM SUN., 31 OCT., 1982

```

0001      FTN4,L,T,C
0002      SUBROUTINE CODNG(MODE,POLAR,A,B,C)
0003      C
0004      IMPLICIT INTEGER(A-Z)
0005      C
0006      *****LABELLED COMMON ARRAYS*****
0007      C
0008      DIMENSION PELREF(258),PELCOD(258),OUTREF(258),OUTCOD(258)
0009      EQUIVALENCE (PELREF,PELBUF),(PELCOD,PELBUF(1,2))
0010      EQUIVALENCE (OUTREF,OTRBUF),(OUTCOD,OTRBUF(1,2))
0011      COMMON/BUFF/PELBUF(258,2),OTRBUF(258,2),STAT(1)
0012      & CDBUF(1024),STFBUF(1024)
0013      COMMON/HUFF/CODE(3,105,2),CODERD(3,11)
0014      COMMON/ERAY/ERRORS(100)
0015      C
0016      *****LABELLED COMMON VARIABLES*****
0017      C
0018      COMMON/IVAR/PELMAX,VRES,EPHASE,CHPMAX,ERRMOD,LINMAX,K
0019      COMMON/PVAR/INLNNO,OTLNNO,OTELV,CDELP,OTELP,CDELV,CDDATA,
0020      CDELCCT,INELCT,TCDATA,TCDEL,ERRPNT,ERROFF,ERRLIM,
0021      ERRCNT,INLNCT,CONSEC,LNNOBF,ZERO,
0022      INCOD,INREF,OTCOD,OTREF,STFBIT,ERRCOR,BUFDIM
0023      COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0024      OUTF
0025      LOGICAL SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0026      OUTF
0027      C
0028      DOUBLE PRECISION TCDATA,TCDEL
0029      C
0030      *****END COMMON*****
0031      C
0032      INTEGER A,B,C,MODE,NEVPOL,POLAR
0033      C
0034      ***** BEGIN PROGRAM *****
0035      C
0036      CALL M128(CODERD(3,MODE),CDBUF,CDELCCT+1,CODERD(1,MODE))
0037      CDELCCT=CDELCCT+CODERD(1,MODE)
0038      GO TO (100,200,100,100,100,100,100,100,800,800),MODE
0039      C
0040      C      MODE      1  2  3  4  5  6  7  8  9 10 11
0041      C
0042      C      PASS MODE(1),VERTICAL MODE:A1B1=B(3),A1B1=1(4,7),=2(5,8),=3(6,9)
0043      C
0044      C      100 CONTINUE
0045      CDDATA=CDDATA+CODERD(1,MODE)
0046      RETURN
0047      C
0048      C      200 CONTINUE
0049      CDDATA=CDDATA+CODERD(1,MODE)
0050      CALL CODLN(B-A,POLAR)
0051      NEVPOL=MOD(POLAR+2,2)+1
0052      CALL CODLN(C-B,NEVPOL)
0053
0054
0055

```

PAGE 0002 CODING 5:03 PM SUN., 31 OCT., 1982

```

0056      RETURN
0057      C
0058      C ADD EOL1 OR EOL2 TO LINE (10,11)
0059      C
0060      000 CONTINUE
0061      RETURN
0062      END
    
```

FTN4 COMPILER: HP92000-16002 REV. 2026 (000423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00112 COMMON = 00000

PAGE 0001 FTM. 5:03 PM SUN., 31 OCT., 1982

```

0001      FTN4,L,T,C
0002      SUBROUTINE CODLN(LENGTH,POLAR)
0003      C
0004      IMPLICIT INTEGER(A-Z)
0005      C
0006      C*****LABELLED COMMON ARRAYS*****
0007      C
0008      DIMENSION PELREF(258),PELCOD(258),OUTREF(258),OUTCOD(258)
0009      EQUIVALENCE (PELREF,PELBUF),(PELCOD,PELBUF(1,2))
0010      EQUIVALENCE (OUTREF,OTBUF),(OUTCOD,OTBUF(1,2))
0011      COMMON/BUFF/PELBUF(258,2),OTBUF(258,2),
0012      & CDBUF(1024),STBUF(1024),STAT(1)
0013      COMMON/HUFF/CODE(3,105,2),CODERD(3,11)
0014      COMMON/ERAY/ERRORS(100)
0015      C
0016      C*****FILE BUFFERS*****
0017      C
0018      COMMON/FILES/TERM,LPFIL,IDC87(144),IDC88(144),IDC89(144)
0019      INTEGER TERM,LPFIL,IDC87,IDC88,IDC89
0020      C
0021      IDC87 - PELFIL
0022      IDC88 - OTFIL
0023      IDC89 - STATFL
0024      C
0025      C*****LABELLED COMMON VARIABLES*****
0026      C
0027      COMMON/IVAR/PELMAX,VRES,EPHASE,CMPMAX,ERRMOD,LINMAX,K
0028      COMMON/PVAR/INLNNO,OTLNNO,OTELV,CDELP,OTELP,CDELV,CDDATA,
0029      & CDELCT,INELCT,TCDDATA,TCDEL,ERRPNT,ERROFF,ERRLIN,
0030      & ERRCNT,INLNCT,CONSEC,LNNBFF,ZERO,
0031      & INCOD,INREF,OTCOD,OTREF,STFBIT,ERRCOR,BUFDIM
0032      COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0033      & OUTF
0034      LOGICAL SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND,
0035      & OUTF
0036      C
0037      DOUBLE PRECISION TCDDATA,TCDEL
0038      C
0039      C*****END COMMON*****
0040      C
0041      INTEGER LENGTH,POLAR,INDEX,DEX,CODESV,TCODE,TLENG
0042      C
0043      C*****BEGIN PROGRAM *****
0044      C
0045      CHECK INPUTS
0046      C
0047      IF(POLAR.LT.1.OR.POLAR.GT.2) STOP 3333
0048      IF(LENGTH.LT.0.OR.LENGTH.GT.PELMAX) STOP 4444
0049      C
0050      IF(LENGTH.LE.63) GO TO 40
0051      C
0052      CALCULATE MAKE UP CODE INDEX
0053      C
0054      INDEX=LENGTH/64+64
0055      20 IF(INDEX.LE.104) GO TO 30

```

PAGE 0002 CODLN 5:03 PM SUN.. 31 OCT.. 1982

```

0056 INDEX=INDEX-40
0057 DEX=104
0058 ASSIGN 20 TO CODESV
0059 IF(DIAG)WRITE(TERM,25)CODESV,DEX,POLAR
0060 D25 FORMAT(1,20 CODESV,"316)
0061 GO TO 1000
0062 30 DEX=INDEX
0063 ASSIGN 40 TO CODESV
0064 IF(DIAG)WRITE(TERM,35)CODESV,DEX,POLAR
0065 D35 FORMAT(1,30 CODESV,"316)
0066 GO TO 1000
0067 C
0068 C CALCULATE TERMINATING CODE INDEX
0069 C
0070 40 DEX=MOD(LENGTH,64)+1
0071 ASSIGN 60 TO CODESV
0072 IF(DIAG)WRITE(TERM,45)CODESV,DEX,POLAR
0073 D45 FORMAT(1,40 CODESV,"316)
0074 GO TO 1000
0075 60 RETURN
0076 C
0077 C CODE LOOK-UP AND INSERTION ROUTINE
0078 C
0079 1000 CONTINUE
0080 TCODE=CODE(3,DEX,POLAR)
0081 TLENG=CODE(1,DEX,POLAR)
0082 IF(DIAG) WRITE(TERM,1005) CODE(1,DEX,POLAR)
0083 D1005 FORMAT(1,WORD LENGTH,"16)
0084 CALL M128(TCODE,CDBUF,CDELCT+1,TLENG)
0085 CDELCT=CDELCT+TLENG
0086 CDDATA=CDDATA+TLENG
0087 C
0088 GO TO CODESV,(20,40,60)
0089 E N D
0090

```

FTM4 COMPILER: NP92060-16092 REV. 2026 (000423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00135 COMMON = 00000

PAGE 0001 FTM. 5:01 PM SUM., 31 OCT., 1982

```

0001      FTM4,L,T,C
0002      SUBROUTINE INOUT(ICTL,IEOF)
0003      IMPLICIT INTEGER(A-Z)
0004
0005      THIS ROUTINE PERFORMS ALL DISC I/O FOR IMAGE FILES
0006
0007      A RECORD CONSISTS OF:
0008      LINE NUMBER - 1 WORD
0009      MAX LINE SIZE - 1 WORD (NEGATIVE FOR A CORRUPTED LINE)
0010      DATA FOR ONE SCAN LINE - UP TO 256 WORDS
0011
0012      THE INPUT/OUTPUT BUFFERS ARE CONNECTED TO THE TWO-DIMENSIONAL
0013      PROGRAM ARRAYS BY EQUIVALENCE. LINE NUMBER AND SIZE MUST BE
0014      LOADED/EXTRACTED BEFORE/AFTER WRITING/READING.
0015
0016      ENTRY PARAMETERS:
0017
0018      ICTL - I/O CONTROL WORD
0019
0020      1- READ INTO REFERENCE BUFFER (PELREF)
0021      2- READ INTO CODE BUFFER (PELCOD)
0022      3- WRITE FROM REFERENCE BUFFER (OUTREF)
0023      4- WRITE FROM CODE BUFFER (OUTCOD)
0024      5- READ INTO REFERENCE BUFFER (OUTREF)
0025      6- READ INTO CODE BUFFER (OUTCOD)
0026      99- CLOSE FILES AND STOP
0027
0028      IEOF - INDICATES POS OR NEG LINE SIZE TO BE WRITTEN
0029      +1 - POS
0030      -1 - NEG
0031
0032      EXIT PARAMETERS:
0033
0034      IEOF - CONTAINS END OF FILE INDICATION
0035
0036      *****LABELLED COMMON ARRAYS*****
0037
0038      DIMENSION PELREF(258),PELCOD(258),OUTREF(258),OUTCOD(258)
0039      EQUIVALENCE (PELREF,PELBUF),(PELCOD,PELBUF(1,2))
0040      EQUIVALENCE (OUTREF,OTBUF),(OUTCOD,OTBUF(1,2))
0041      COMMON/BUFF/PELBUF(258,2),OTBUF(258,2),
0042      & CDBUF(1024),STFBUF(1024),STAT(1)
0043      COMMON/HUFF/CODE(3,105,2),CODERD(3,11)
0044      COMMON/ERAY/ERRORS(100)
0045
0046      *****FILE BUFFERS*****
0047
0048      COMMON/FILES/TERM,LPFIL,IDCB7(144),IDCB8(144),IDCB9(144)
0049      INTEGER TERM,LPFIL,IDCB7,IDCB8,IDCB9
0050
0051      IDCB7 - PELFIL
0052      IDCB8 - OTFIL
0053      IDCB9 - STATFL
0054
0055

```

PAGE 0002 INOUT 5:01 PM SUN.. 31 OCT.. 1982

```

0056 C*****LABELLED COMMON VARIABLES*****
0057 C
0058 COMMON/IVAR/PELMAX,VRES,EPHASE,CHPMAX,ERRMOD,LINMAX,K
0059 COMMON/PVAR/INLNNO,OTLNNO,OTELV,CDELP,OTELP,CDELV,CDDATA,
0060 CDELCI,INELCT,TCDDATA,TCDEL,ERRPNT,ERROFF,ERRLIN,
0061 ERRCNT,INLNCT,CONSEC,INNOBF,ZERO,
0062 INCOD,INREF,OTCOD,OTREF,STF8IT,ERRCOR,BUFDIM
0063 COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,LEFT,CNCOL,ONE,WHITE,FILEND,
0064 OUTF
0065 LOGICAL SEARCH,DIAG,SYNC,WRITE,LEFT,CNCOL,ONE,WHITE,FILEND,
0066 OUTF
0067 C
0068 C DOUBLE PRECISION TCDDATA,TCDEL
0069 C
0070 C*****END COMMON*****
0071 C
0072 C*****BEGIN PROGRAM*****
0073 C
0074 C GO TO(100,200,300,400,500,600,9900),ICTL
0075 C
0076 C READ INTO REFERENCE BUFFER
0077 C
0078 C 100 CALL READF(IDC87,IERR,PELREF,BUFDIM+2,IEOF)
0079 C IF(.NOT.OUTF) GO TO 9000
0080 C INLNNO=PELREF(BUFDIM+1)
0081 C INELCT=PELREF(BUFDIM+2)
0082 C GO TO 9000
0083 C
0084 C READ INTO CODE BUFFER
0085 C
0086 C 200 CALL READF(IDC87,IERR,PELCOD,BUFDIM+2,IEOF)
0087 C IF(.NOT.OUTF) GO TO 9000
0088 C INLNNO=PELCOD(BUFDIM+1)
0089 C INELCT=PELCOD(BUFDIM+2)
0090 C GO TO 9000
0091 C
0092 C WRITE FROM REFERENCE BUFFER
0093 C
0094 C 300 OUTREF(BUFDIM+1)=OTLNNO
0095 C OUTREF(BUFDIM+2)=IEOF*PELMAX
0096 C CALL WRITF(IDC88,IERR,OUTREF)
0097 C GO TO 1500
0098 C
0099 C WRITE FROM CODE BUFFER
0100 C
0101 C 400 OUTCOD(BUFDIM+1)=OTLNNO
0102 C OUTCOD(BUFDIM+2)=IEOF*PELMAX
0103 C CALL WRITF(IDC88,IERR,OUTCOD)
0104 C GO TO 1500
0105 C
0106 C READ FROM OUTPUT REFERENCE BUFFER
0107 C
0108 C 500 CALL READF(IDC88,IERR,OUTREF,BUFDIM+2,IEOF)
0109 C OTLNNO=OUTREF(BUFDIM+1)
0110 C LOCAL -OUTREF(BUFDIM+2)

```

PAGE 0003 INOUT 5:01 PM SUN.. 31 OCT.. 1982

```
0111      GO TO 9000
0112      C
0113      C READ FROM OUTPUT CODE BUFFER
0114      C
0115      600 CALL READF(IDC88,IERR,OUTCOD,BUFDIM+2,IEOF)
0116      OTLNNO=OUTCOD(BUFDIM+1)
0117      LOCAL -OUTCOD(BUFDIM+2)
0118      GO TO 9000
0119      C
0120      C TEST FOR ERROR
0121      C
0122      1500 CONTINUE
0123      IEOF=-1
0124      IF(IERR.EQ.-12) IEOF=-1
0125      IF(IEOF.EQ.-1) RETURN
0126      IF(IERR.GE.0) RETURN
0127      WRITE(TERM,9100) IERR
0128      9100 FORMAT('FILE ERROR-16'. ABORT1')
0129      CALL CLOSE(IDC87,IERR)
0130      IF(.NOT.OUTF) GO TO 9940
0131      CALL LOCF(IDC88,IERR,1,IRB,1,JSEC)
0132      ITRUN=JSEC/2-IRB-1
0133      CALL CLOSE(IDC88,IERR,ITRUN)
0134      IF(IERR.LT.0) STOP 1500
0135      9940 CONTINUE
0136      CALL CLOSE(IDC89,IERR)
0137      WRITE(TERM,9950)
0138      9950 FORMAT('BRUN COMPLETE1')
0139      STOP
0140      END
```

FTN4 COMPILER, HP9200B-16092 REV. 2026 (800423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00255 COMMON = 00000

APPENDIX B
CODE LISTING FOR THE MODIFIED READ
CODE II WITH WRAP-AROUND

<u>SUBROUTINE</u>	<u>PAGE NO.</u>
1. MRCW	B-1
2. INITW	B-3
3. ENCDW	B-7
4. GTRAN	B-12
5. CODNG	B-15
6. CODLN	B-17

PAGE 0001 FTH. 4:56 PM SUN., 31 OCT., 1982

```

0001      FTH4,L.T.C
0002      PROGRAM MRCV
0003      C
0004      IMPLICIT INTEGER(A-Z)
0005      C
0006      ASSUMPTIONS:
0007      MAXIMUM LINE LENGTH=4096
0008      MAXIMUM NUMBER OF LINES=5600
0009      MAXIMUM INPUT RECORD SIZE=256
0010      C
0011      C*****LABELLED COMMON ARRAYS*****
0012      C
0013      COMMON/BUF/PELBUF(250),CDBUF(1024),STAT(1),TRANS(3,1024)
0014      COMMON/HUFF/CODE(3,105,2),CODERD(3,11)
0015      C
0016      C*****FILE BUFFERS*****
0017      C
0018      COMMON/FILES/TERM,LPFIL,IDCB7(144),IDCB8(144),IDCB9(144)
0019      INTEGER TERM,LPFIL,IDCB7,IDCB8,IDCB9
0020      C
0021      IDCB7 - PELFIL
0022      IDCB8 - OTFIL
0023      IDCB9 - STATFL
0024      C
0025      C*****LABELLED COMMON /G16BT/ *****
0026      C
0027      DIMENSION MASK(16),COMASK(16),LIBIT(16),LZBIT(16)
0028      COMMON /G16BT/MASK,COMASK,LIBIT,LZBIT,NBPV
0029      C
0030      C*****LABELLED COMMON VARIABLES*****
0031      C
0032      COMMON/IVAR/PELMAX,LINMAX,K
0033      COMMON/PVAR/INLNMNO,INLNCT,CDELCT,ICDATA,TCDEL,CDELV,CODATA,
0034      STFBIT,BUFDIM,TOP,BOT,TOPREF,COLOR,INP,PHXD64
0035      COMMON/LOGIC/DIAG,FILEND,OUTF,DONE,WRAP
0036      LOGICAL DIAG,FILEND,OUTF,DONE,WRAP
0037      C
0038      DOUBLE PRECISION TCDATA,TCDEL
0039      C
0040      C*****END COMMON*****
0041      C
0042      C*****LOCAL VARIABLES*****
0043      C
0044      REAL CF3,CF4
0045      C
0046      CALL INITV
0047      100 CONTINUE
0048      IF(INLNMNO.GT.LINMAX) GO TO 2000
0049      CALL ENCDV
0050      INLNMNO=INLNMNO+1
0051      GO TO 100
0052      2000 CONTINUE
0053      C
0054      C REPORT COMPRESSION FACTOR, ERROR SENSITIVITY FACTOR
0055      C

```

PAGE 0002 MRCV 4:56 PM SUN.. 31 OCT.. 1982

```

0056      ERRATE=FLOAT(ERRCNT)/TCDEL
0057      C
0058      CALL FTIME(PELBUF)
0059      WRITE(LPFILE,4000) (PELBUF(I),I=1,16)
0060      WRITE(LPFILE,2020) TCDEL,TCDATA
0061      2020 FORMAT('CODED BITS = ',F8.0)
0062      C
0063      C
0064      CF3=FLOAT(PELMAX)*FLOAT(INLNCT)/TCDEL
0065      CF4=FLOAT(PELMAX)*FLOAT(INLNCT)/TCDATA
0066      C
0067      WRITE(LPFILE,2030) CF3,CF4
0068      2030 FORMAT('CF3 = ',F8.4,
0069                ' CF4 = ',F8.4)
0070      C
0071      C
0072      C
0073      C
0074      C
0075      C
0076      C
0077      C
0078      C
0079      C
0080      C
0081      C
0082      C
0083      C
0084      C

```

FTN4 COMPILER: HP92060-16092 REV. 2026 (000423)

COMMON = 00000

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00235

PAGE 0001 FTN. 4:57 PM SUN., 31 OCT., 1902

```

0001 FTN4.L.T.C
0002 SUBROUTINE INITV
0003 C
0004 C IMPLICIT INTEGER(A-Z)
0005 C
0006 C *****LABELLED COMMON ARRAYS*****
0007 C
0008 C COMMON/BUFF/PELBUF(258),CDBUF(1024),STAT(1),TRANS(3,1024)
0009 C COMMON/HUFF/COEF(3,105,2),CODERD(3,11)
0010 C
0011 C *****FILE BUFFERS*****
0012 C
0013 C COMMON/FILES/TERM,LPFIL,IDCB7(144),IDCB8(144),IDCB9(144)
0014 C INTEGER TERM,LPFIL,IDCB7,IDCB8,IDCB9
0015 C
0016 C IDC87 - PELFIL
0017 C IDC88 - OTFIL
0018 C IDC89 - STATFL
0019 C
0020 C *****LABELLED COMMON /G16BT/ *****
0021 C
0022 C DIMENSION MASK(16),COMASK(16),LIBIT(16),LZBIT(16)
0023 C COMMON /G16BT/MASK,COMASK,LIBIT,LZBIT,NBPV
0024 C
0025 C *****LABELLED COMMON VARIABLES*****
0026 C
0027 C COMMON/IVAR/PELMAX,LINMAX,K
0028 C COMMON/PVAR/INLNCT,CDELCT,TCDATA,TCDEL,CDELW,CDDATA,
0029 C STFBIT,BUFDIM,TOP,BOT,TOPREF,COLOR,INP,PMXD64
0030 C COMMON/LOGIC/DIAG,FILEND,OUTF,DONE,WRAP
0031 C LOGICAL DIAG,FILEND,OUTF,DONE,WRAP
0032 C
0033 C DOUBLE PRECISION TCDEL
0034 C
0035 C *****END COMMON*****
0036 C
0037 C *****FILE PARAMETERS*****
0038 C
0039 C DIMENSION LINE(80),IBUF(10),ITBUF(15),NMBUF(3)
0040 C
0041 C EQUIVALENCE(IBUF,FNAM),(IBUF(5),ISECU),(IBUF(6),ICR)
0042 C
0043 C DATA LINE/80*2H /,LEN/80/,ISTRC/1/
0044 C
0045 C ***** BEGIN PROGRAM *****
0046 C
0047 C
0048 C GET INPUT IMAGE FILE NAME AND OPEN
0049 C
0050 C CALL GETST(LINE,LEN,IDUM)
0051 C CALL NMR(IBUF,LINE,2*IDUM,ISTRC)
0052 C CALL OPEN(IDCB7,IERR,FNAM,B,ISECU,ICR)
0053 C IF(IERR.LT.B)STOP 0001
0054 C
0055 C GET OUTPUT IMAGE FILE NAME AND OPEN

```

PAGE 0002 INITV 4:57 PM SUN.. 31 OCT.. 1982

```

0056 C          CALL NAMR(1BUF,LINE,2*10UM,ISTRC)
0057          CALL OPEN(1DC89,IERR,FNAM,0,ISECU,ICR)
0058          IF(IERR,GE,0) GO TO 5
0059 *****
0060 *****
0061 C          IF NO OUTPUT FILE
0062          THEN NO ERROR INSERTION
0063          AND
0064          NO ERROR COMPARISON (ERRMS)
0065          AND
0066          NO INPUT LINE NUMBER OR PEL COUNT USED/REQUIRED
0067          ELSE ERROR INSERTION OPTIONAL
0068 *****
0069 *****
0070 C          NO OUTPUT FILE
0071 C
0072 C
0073 C
0074          OUTF=FALSE.
0075          WRITE(TERM,6)
0076          6 FORMAT(' NO OUTPUT FILE SPECIFIED.')
0077 C
0078 C          GET SCRATCH FILE NAME & OPEN
0079 C
0080          5 CALL NAMR(1BUF,LINE,2*10UM,ISTRC)
0081          CALL OPEN(1DC89,IERR,FNAM,0,ISECU,ICR)
0082          IF(IERR,LT,0) STOP 0003
0083 C
0084 C          READ INPUT RECORD SIZE
0085 C
0086          20 WRITE(TERM,30)
0087          30 FORMAT(' ENTER INPUT RECORD SIZE: ')
0088          READ(TERM,*) BUFDIM
0089          IF(BUFDIM,GE,0.AND,BUFDIM,LE,256) GO TO 114
0090          WRITE(TERM,150) BUFDIM
0091          GO TO 20
0092 C
0093 C          READ DIAGNOSTIC SWITCH
0094 C
0095          114 WRITE(TERM,115)
0096          115 FORMAT(' DIAGNOSTIC PRINTOUT? (Y OR N): ')
0097          READ(TERM,110) INSW
0098          110 FORMAT(A1)
0099          IF(INSW,EO,2HY) GO TO 116
0100          IF(INSW,EO,2HH) GO TO 120
0101          GO TO 114
0102          116 CONTINUE
0103          DIAG=TRUE.
0104 C
0105 C          READ MAXIMUM NUMBER OF PELS PER LINE
0106 C
0107          120 CONTINUE
0108          WRITE(TERM,130)
0109          130 FORMAT(' ENTER MAXIMUM NUMBER OF PELS PER LINE: ')
0110          READ(TERM,*) PELMAX

```

PAGE 0003 INITV 4:57 PM SUN., 31 OCT., 1982

```

0111 140 FORMAT(14)
0112 IF(MOD(PELMAX,64).NE.0) GO TO 145
0113 IF(PELMAX.GE.1.AND.PELMAX.LE.4896) GO TO 320
0114 145 WRITE(TERM,150) PELMAX
0115 150 FORMAT("NUMBER OUT OF RANGE (-,16,)")
0116 GO TO 120
0117 C
0118 C READ NUMBER OF SCAN LINES TO BE PROCESSED
0119 C
0120 320 CONTINUE
0121 WRITE(TERM,330)
0122 330 FORMAT("NUMBER OF SCAN LINES TO BE PROCESSED=? ")
0123 READ(TERM,*) LINMAX
0124 IF(LINMAX.GE.1.AND.LINMAX.LE.5600) GO TO 350
0125 WRITE(TERM,150) LINMAX
0126 GO TO 320
0127 350 CONTINUE
0128 C
0129 C READ INPUT IMAGE NAME
0130 C
0131 WRITE(TERM,360)
0132 360 FORMAT(" ENTER INPUT IMAGE NAME:")
0133 READ(TERM,365) NMBUF
0134 365 FORMAT(3A2)
0135 C WRITE INPUT PARAMETERS
0136 CALL FTIME(ITBUF)
0137 WRITE(LPFILE,370) ITBUF
0138 370 FORMAT(1H0,15A2)
0139 WRITE(LPFILE,380) NMBUF
0140 380 FORMAT("IMAGE NAME - '3A2'")
0141 C
0142 WRITE(LPFILE,400) PELMAX,LINMAX,BUFDIM
0143 400 FORMAT("INPUT PARAMETERS:")
0144 * - MAXIMUM NUMBER OF PELS PER LINE=",16/
0145 * - NUMBER OF SCAN LINES TO BE PROCESSED "=",16/
0146 * - RECORD SIZE ="14)
0147 WRITE(LPFILE,410)
0148 410 FORMAT("NO ERRORS INSERTED")
0149 C-----***** BEGIN PROGRAM *****
0150 C
0151 C INITIALIZE
0152 C
0153 WRAP=.TRUE.
0154 INLNO=1
0155 PHX064=PELMAX/64
0156 INELCT=PELMAX
0157 CDELT=NBPV
0158 CDELP=NBPV+1
0159 DO 000 I=1,BUFDIM*4
0160 C08UF(I)=0
0161 000 CONTINUE
0162 DO 050 I=1,BUFDIM
0163 PELBUF(I)=0
0164 050 CONTINUE
0165 C

```

PAGE 0004 INITV 4:57 PM SUN., 31 OCT., 1982

```
0166 C C FILL TRANSITIONS LIST
0167 C
0168 DONE=FALSE.
0169 TOP=1
0170 BOT=1
0171 INP=0
0172 DO 1000 I=1,1020
0173 IF(DONE) GO TO 1010
0174 CALL GTRAN
0175 IF(BOT.EQ.TOP) STOP 1000
0176 1000 CONTINUE
0177 1010 CONTINUE
0178 RETURN
0179 E N D
```

FTN4 COMPILER: HP9200B-16092 REV. 2026 (000423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00729 COMMON = 00000

PAGE 0001 FTN. 4:56 PM SUN., 31 OCT., 1982

```

0001      FTN4,L,T,C
0002      SUBROUTINE ENCDW
0003      C
0004      IMPLICIT INTEGER(A-Z)
0005      C
0006      C*****LABELLED COMMON ARRAYS*****
0007      C
0008      COMMON/BUFF/PELBUF(258),CDBUF(1024),STAT(1),TRANS(3,1024)
0009      COMMON/HUFF/CODE(3,105,2),CODERD(3,11)
0010      C
0011      C*****FILE BUFFERS*****
0012      C
0013      COMMON/FILES/TERM,LPFIL,IDCB7(144),IDCB8(144),IDCB9(144)
0014      INTEGER TERM,LPFIL,IDCB7,IDCB8,IDCB9
0015      C
0016      IDC87 - PELFIL
0017      IDC88 - OTFIL
0018      IDC89 - STATFL
0019      C
0020      C*****LABELLED COMMON /G16BT/ *****
0021      C
0022      DIMENSION MASK(16),CONMASK(16),LIBIT(16),LZBIT(16)
0023      COMMON /G16BT/MASK,CONMASK,LIBIT,LZBIT,NBPV
0024      C
0025      C*****LABELLED COMMON VARIABLES*****
0026      C
0027      COMMON/IVAR/PELMAX,LINMAX,K
0028      COMMON/PVAR/INLNNO,INLNCT,CDELCT,TCDATA,TCDEL,CDELV,CODATA,
0029      STFBIT,BUFDIM,TOP,BOT,TOPREF,COLOR,INP,PHXD64
0030      COMMON/LOGIC/DIAG,FILEND,OUTF,DONE,WRAP
0031      LOGICAL DIAG,FILEND,OUTF,DONE,WRAP
0032      C
0033      DOUBLE PRECISION TCDATA,TCDEL
0034      C
0035      C*****END COMMON*****
0036      C
0037      C***** BEGIN PROGRAM *****
0038      C
0039      INCMO(IX)=MOD(IX+1024,1024)+1
0040      DIFF(IA,IB,ILA,ILB)=(ILA-ILB-1)*PELMAX+IA-IB
0041      DIFF(IA,IB,ILA,ILB)=(ILA-ILB+1)*PELMAX+IA-IB
0042      C
0043      C INITIALIZE VARIABLES
0044      C
0045      CDELCT=NBPV
0046      CODATA=0
0047      DO 50 I=2,BUFDIM*4
0048      CDBUF(I)=0
0049      50 CONTINUE
0050      C
0051      IF(INLNNO.NE.1) GO TO 600
0052      C
0053      ONE-DIMENSIONAL CODING
0054      WRITE ONE EOL
0055      C

```

PAGE 0002 ENCDV 4:56 PM SUN., 31 OCT., 1982

```

0056 CALL CODNG(10,B)
0057 T81-TOP
0058 POLAR=1
0059 A=1
0060 LA=1
0061 LB=1
0062 C
0063 C CALCULATE RUNLENGTH AND ENCODE
0064 C
0065 150 CONTINUE
0066 B-TRANS(2,TOP)
0067 C
0068 C TEST FOR END OF LINE
0069 C
0070 IF(TRANS(1,TOP).GT.INLNNO) B=PELMAX+1
0071 CALL CODLN(B-A,LB-LA,POLAR)
0072 RUN=B-A
0073 IF(01AG)WRITE(TERM,160) RUN,POLAR,CDELCT,CDDATA
0074 160 FORMAT(818)
0075 C
0076 C UPDATE OLD RUN END AND POLARITY
0077 C
0078 A=B
0079 POLAR=TRANS(3,TOP)+1
0080 C
0081 C TEST FOR END OF LINE
0082 C
0083 IF(A.GT.PELMAX) GO TO 1000
0084 C
0085 C INCREMENT TOP
0086 C
0087 TOP=INCHD(TOP)
0088 IF(TOP.EQ.BOT) STOP 160
0089 IF(.NOT.DONE.AND.INCHD(BOT).NE.T81) CALL GTRAN
0090 GO TO 150
0091 C
0092 C TWO-DIMENSIONAL CODING
0093 C
0094 600 CONTINUE
0095 STFBIT-STFBIT+1
0096 C
0097 C IF PREVIOUS LINE IS ONE-DIMENSIONAL, WRITE ONE EOL2
0098 C
0099 CALL CODNG(11,B)
0100 C
0101 C SET AB TO LEFT EDGE-1
0102 C
0103 A0=B
0104 LAB=2
0105 TAI-TOP
0106 POLAR=1
0107 GO TO 620
0108 C
0109 C INITIALIZE CODE LINE POINTERS
0110 C

```

PAGE 0553 ENCDV 4:56 PM SUN., 31 OCT., 1982

```

0111 61# CONTINUE
0112 CDELCT-NBPV
0113 CDDATA-B
0114 DO 615 I=2,8UFDM*4
0115 CD8UF(1)-B
0116 615 CONTINUE
0117 C
0118 C DETECT A1
0119 C
0120 62# CONTINUE
0121 A1-TRANS(2,TA1)
0122 LA1-TRANS(1,TA1)
0123 IF(LA1.GT.LINMAX) LA1=LINMAX
0124 C
0125 C DETECT B1
0126 C
0127 65# CONTINUE
0128 B1-TRANS(2,TB1)
0129 LB1-TRANS(1,TB1)
0130 IF(B1.GT.PELMAX) GO TO 7#B
0131 IF(LB1.GE.LAB) GO TO 65#
0132 IF(LB1+1.EQ.LAB.AND.B1.GT.AB) GO TO 65#
0133 GO TO 66#
0134 C
0135 C B1 TO RIGHT OF AB TEST FOR OPPOSITE POLARITY
0136 C
0137 65# IF(TRANS(3,TB1)+1.NE.POLAR) GO TO 67#
0138 C
0139 C SAME COLOR LOOK AT NEXT TRANSITION
0140 C
0141 66# TB1=INCHD(TB1)
0142 GO TO 65#
0143 C
0144 C HAVE B1
0145 C
0146 67# CONTINUE
0147 C
0148 C DETECT B2
0149 C
0150 TB2=INCHD(TB1)
0151 B2-TRANS(2,TB2)
0152 LB2-TRANS(1,TB2)
0153 IF(LB2.GT.PELMAX) GO TO 71#
0154 GO TO 74#
0155 7#B B1=PELMAX+1
0156 LB1=LINMAX
0157 71# B2=PELMAX+1
0158 LB2=LINMAX
0159 74# CONTINUE
0160 C
0161 C TEST FOR PASS MODE
0162 C
0163 IF(LB2.GE.LA1) GO TO 75#
0164 IF(LB2+1.EQ.LA1.AND.B2.GE.A1) GO TO 75#
0165 C

```

PAGE 8884 ENCDW 4:56 PM SUN., 31 OCT., 1982

```

8166 C PASS MODE CODING (CAN'T END A LINE IN PASS MODE; NEW AB MUST HAVE
8167 C SAME POLARITY AS B2)
8168 C
8169 CALL CODNG(1,8)
8170 AB=B2
8171 LAB=L82+1
8172 TB1-TB2
8173 GO TO 62B
8174 75B CONTINUE
8175 C
8176 IF (IABS(LA1-(LB1+1)).GT.1) GO TO 799
8177 MAB-IABS(DIFF(A1,B1,LA1,LB1))
8178 IF (DIAG) WRITE(LPFIL,16B) A1,B1,LA1,LB1,MAB
8179 IF (MAB-3) 835,835,799
8180 C
8181 C CODE BY HORIZONTAL MODE; FIRST DETECT A2
8182 C
8183 799 CONTINUE
8184 IF (A1.GT.PELMAX) GO TO 88B
8185 TA2=INCHD(TA1)
8186 A2=TRANS(2,TA2)
8187 LA2=TRANS(1,TA2)
8188 IF (LA2.GT.LINMAX) LA2=LINMAX
8189 GO TO 81B
8190 88B CONTINUE
8191 A2=PELMAX+1
8192 LA2=LA1
8193 TA2=TA1
8194 81B CONTINUE
8195 IF (INLNO.EQ.2.AND.AB.EQ.8) AB=1
8196 CALL CODNG(2,POLAR)
8197 IF (DIAG) WRITE(LPFIL,16B) AB,A1,A2,LAB,LA1,LA2,POLAR
8198 CALL CODLN(A1-AB,LA1-LAB,POLAR)
8199 NEWPOL=MOD(POLAR+2,2)+1
8200 CALL CODLN(A2-A1,LA2-LA1,NEVPOL)
8201 AB=A2
8202 LAB=LA2
8203 TOP=TA2
8204 IF (.NOT.DONE.AND.INCHD(BOT).NE.TB1) CALL STRAN
8205 IF (.NOT.DONE.AND.INCHD(BUT).NE.TB1) CALL STRAN
8206 GO TO 96B
8207 C
8208 C CODE BY VERTICAL MODE
8209 C
8210 835 CONTINUE
8211 AIMB1=DIFF(A1,B1,LA1,LB1)
8212 IF (AIMB1) 85B,84B,84B
8213 C
8214 84B CALL CODNG(AIMB1+3,8)
8215 GO TO 95B
8216 85B CONTINUE
8217 B1MA1=DIFF(B1,A1,LB1,LA1)
8218 CALL CODNG(B1MA1+5,8)
8219 95B CONTINUE
8220 AB=A1

```

PAGE 0005 ENCDV 4:56 PM SUN.. 31 OCT.. 1982

```

0221 LAB-LAI
0222 TOP-TAI
0223 IF(.NOT.DONE.AND.INCMD(BOT).NE.TBI) CALL GTRAN
0224 C
0225 C TEST FOR END OF LINE
0226 C
0227 960 CONTINUE
0228 POLAR-TRANS(3, TOP)+1
0229 TAI=INCMD(TOP)
0230 IF(TRANS(1, TOP).LE.INLMO) GO TO 620
0231 C
0232 C LINE FINISHED
0233 C
0234 INLMO-TRANS(1, TOP)
0235 1000 CONTINUE
0236 C SAVE LINE LENGTH (DATA BITS ONLY)
0237 C
0238 STAT(1)=CDDATA
0239 CALL WRITF(1DCB9, IERR, STAT)
0240 IF(IERR.LT.0) STOP 300
0241 C
0242 C COMPUTE STATISTICS
0243 C
0244 300 CONTINUE
0245 TCDEL=TCDEL+CDELCT-NBPV
0246 TCDDATA=TCDDATA+CDDATA
0247 C
0248 IF (.NOT.DIAG) GO TO 460
0249 CDELV=(CDELCT+NBPV-1)/NBPV
0250 WRITE(LPFILE, 450) (CDBUF(1), 1-1, CDELV)
0251 450 FORMAT(80I2)
0252 460 CONTINUE
0253 C
0254 C TEST FOR END OF FILE
0255 C
0256 IF(INLMO.GT.1.AND.INLMO.LT.32000) GO TO 610
0257 RETURN
0258 C
0259 C
0260 E N D

```

FTM4 COMPILER: HP92060-16092 REV. 2026 (800423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00040 COMMON = 00000

00040

PAGE 0001 FTM. 4:57 PM SUN., 31 OCT., 1982

```

0001      FTM4,L,T,C
0002      SUBROUTINE GTRAN
0003
0004      C RETURNS ONE TRANSITION AT EACH CALL
0005      C
0006      C IMPLICIT INTEGER(A-Z)
0007
0008      C*****LABELLED COMMON ARRAYS*****
0009
0010      COMMON/BUFF/PELBUF(258),CDBUF(1024),STAT(1),TRANS(3,1024)
0011      COMMON/HUFF/CODE(3,105,2),CODERD(3,11)
0012
0013      C*****FILE BUFFERS*****
0014
0015      COMMON/FILES/TERM,LPFIL,IDC87(144),IDC88(144),IDC89(144)
0016      INTEGER TERM,LPFIL,IDC87,IDC88,IDC89
0017
0018      C IDC87 - PELFIL
0019      C IDC88 - OTFIL
0020      C IDC89 - STATFL
0021
0022      C*****LABELLED COMMON /G16BT/ *****
0023
0024      DIMENSION MASK(16),COMASK(16),LIBIT(16),LZBIT(16)
0025      COMMON /G16BT/MASK,COMASK,LIBIT,LZBIT,NBPV
0026
0027      C*****LABELLED COMMON VARIABLES*****
0028
0029      COMMON/IVAR/PELMAX,LINMAX,K
0030      COMMON/PVAR/INLNNO,INLNCT,CDELCT,TCDATA,TCDEL,CDELW,CDDATA,
0031      * STFBIT,BUFDIM,TOP,BOT,TOREF,COLOR,INP,PMXD64
0032      COMMON/LOGIC/DIAG,FILEND,OUTF,DONE,WRAP
0033      LOGICAL DIAG,FILEND,OUTF,DONE,WRAP
0034
0035      C DOUBLE PRECISION TCDATA,TCDEL
0036
0037      C*****END COMMON*****
0038
0039      INCMD(IX)=MOD(IX+1024,1024)+1
0040
0041      C***** BEGIN PROGRAM *****
0042
0043      C LINE AVAILABLE?
0044
0045      100 CONTINUE
0046      IF(1HP) 110,200,300
0047      110 STOP 100
0048
0049      C NO READ NEXT SCAN LINE
0050
0051      200 CONTINUE
0052      CALL READK(IDC87,IERR,PELBUF,BUFDIM,LEN)
0053      IF(IERR.EQ.-12.OR.LEN.EQ.-1) GO TO 250
0054      IF(IERR.LT.0) STOP 200
0055      IF(INLNCT.GE.LINMAX) GO TO 250

```

PAGE 0002 GTRAN 4:57 PM SUN.. 31 OCT.. 1982

```

0056 INLNCT=INLNCT+1
0057 IF(WRAP) GO TO 260
0058
0059 C STORE REFERENCE TRANSITION
0060 C
0061 TRANS(1,BOT)=INLNCT
0062 TRANS(2,BOT)=0
0063 TRANS(3,BOT)=0
0064 IF(DIAG)WRITE(LPFIL,500)TRANS(1,BOT),TRANS(2,BOT),TRANS(3,BOT),BOT
0065 BOT=INCHD(BOT)
0066 COLOR=0
0067 INP=1
0068 RETURN
0069
0070 C EOF STORE FINAL TRANSITION WITH MAX LINE NO. AT 0 AND WHITE
0071 C SET DONE FLAG
0072 C
0073 C 250 CONTINUE
0074 DONE=.TRUE.
0075 TRANS(1,BOT)=32000
0076 TRANS(2,BOT)=PELMAX+1
0077 TRANS(3,BOT)=0
0078 IF(DIAG)WRITE(LPFIL,500)TRANS(1,BOT),TRANS(2,BOT),TRANS(3,BOT),BOT
0079 BOT=INCHD(BOT)
0080 RETURN
0081 260 INP=1
0082
0083 C LINE AVAILABLE TEST FOR BEGINNING OF WORD
0084 C
0085 300 CONTINUE
0086 IF(INP.GT.PELMAX) GO TO 700
0087 IF(MOD(INP-1,NBPV)) 110,350,400
0088
0089 C AT BEGINNING OF WORD TEST FOR COLOR CHANGE
0090 C
0091 350 CONTINUE
0092 PEL=140(PELBUF,INP,1)
0093 IF(PEL.NE.COLOR) GO TO 420
0094 WORD=(INP+NBPV-1)/NBPV
0095 GO TO (360,370).COLOR+1
0096 360 IF(PELBUF(WORD)) 400,300,400
0097 370 IF(PELBUF(WORD).NE.-1) GO TO 400
0098
0099 C WORD ALL BLACK OR ALL WHITE
0100 C
0101 380 CONTINUE
0102 INP=INP+16
0103 IF(INP.LE.PELMAX) GO TO 350
0104 GO TO 700
0105
0106 C NOT AT BEGINNING OF WORD OR NOT ALL ONE COLOR
0107 400 CONTINUE
0108 PEL=140(PELBUF,INP,1)
0109 IF(PEL.NE.COLOR) GO TO 420
0110 INP=INP+1

```

PAGE 0003 STRAN 4:57 PM SUN., 31 OCT., 1982

```

0111 IF(IMP.LE.PELMAX) GO TO 300
0112 GO TO 700
0113 C HAVE A TRANSITION
0114 C
0115 C 420 CONTINUE
0116 COLOR=PEL
0117 TRANS(1,BOT)=INLCT
0118 TRANS(2,BOT)=IMP
0119 TRANS(3,BOT)=COLOR
0120 IF(DIAG)WRITE(LPFILE,500)TRANS(1,BOT),TRANS(2,BOT),TRANS(3,BOT),BOT
0121 500 FORMAT(410)
0122 C
0123 C INCREMENT BOT POINTER
0124 C
0125 C BOT-INCHD(BOT)
0126 IMP=IMP+1
0127 RETURN
0128 C
0129 C ADD END-OF-LINE TRANSITION
0130 C
0131 C 700 CONTINUE
0132 IF(WRAP) GO TO 200
0133 TRANS(1,BOT)=INLCT
0134 TRANS(2,BOT)=PELMAX+1
0135 TRANS(3,BOT)=0
0136 IMP=0
0137 IF(DIAG)WRITE(LPFILE,500)TRANS(1,BOT),TRANS(2,BOT),TRANS(3,BOT),BOT
0138 BOT-INCHD(BOT)
0139 RETURN
0140 E N D
0141

```

FTM4 COMPILER: HP32060-16092 REV. 2026 (000423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00516 COMMON = 00000

PAGE 0001 FTM. 4:57 PM SUN.. 31 OCT.. 1982

```

0001      FTM4,L,T,C
0002      SUBROUTINE CODING(MODE,POLAR)
0003      C
0004      IMPLICIT INTEGER(A-Z)
0005      C
0006      C*****LABELLED COMMON ARRAYS*****
0007      C
0008      COMMON/BUFF/PELBUF(258),CDBUF(1024),STAT(1),TRANS(3,1024)
0009      COMMON/HUFF/CODE(3,105,2),CODERD(3,11)
0010      C
0011      C*****FILE BUFFERS*****
0012      C
0013      COMMON/FILES/TERM,LPFIL,IDCB7(144),IDCB8(144),IDCB9(144)
0014      INTEGER TERM,LPFIL,IDCB7,IDCB8,IDCB9
0015      C
0016      IDC87 - PELFIL
0017      IDC88 - OTFIL
0018      IDC89 - STATFL
0019      C
0020      C*****LABELLED COMMON /G16BT/ *****
0021      C
0022      DIMENSION MASK(16),COMASK(16),LIBIT(16),LZBIT(16)
0023      COMMON /G16BT/MASK,COMASK,LIBIT,LZBIT,NBPW
0024      C
0025      C*****LABELLED COMMON VARIABLES*****
0026      C
0027      COMMON/IVAR/PELMAX,LINMAX,K
0028      COMMON/PVAR/INLWNO,INLNCT,CDELCT,TCDATA,TCDEL,CDELW,CDDATA,
0029      * STFBIT,BUFDIM,TOP,BOT,TOPREF,COLOR,INP,PMXD64
0030      COMMON/LOGIC/DIAG,FILEND,OUTF,DONE,WRAP
0031      LOGICAL DIAG,FILEND,OUTF,DONE,WRAP
0032      C
0033      DOUBLE PRECISION TCDEL
0034      C
0035      C*****END COMMON*****
0036      C
0037      INTEGER A,B,C,MODE,NEWPOL,POLAR
0038      C
0039      C***** BEGIN PROGRAM *****
0040      C
0041      CALL M128(CODERD(3,MODE),CDBUF,CDELCT+1,CODERD(1,MODE))
0042      CDELCT=CDELCT+CODERD(1,MODE)
0043      GO TO (100,200,100,100,100,100,100,100,100,800,800), MODE
0044      C
0045      C      MODE      1  2  3  4  5  6  7  8  9 10 11
0046      C
0047      C      PASS MODE(1),VERTICAL MODE,A181-B(3),A181-1(4,7),-2(5,8),-3(5,9)
0048      C
0049      100 CONTINUE
0050      CDDATA=CDDATA+CODERD(1,MODE)
0051      RETURN
0052      C
0053      C      HORIZONTAL MODE(2)
0054      C
0055      200 CONTINUE

```

PAGE 0002 CODING 4:57 PM SUN., 31 OCT., 1982

```

0056 CDDATA-CDDATA+COORD(1,MODE)
0057 RETURN
0058 C
0059 C ADD EOL1 OR EOL2 TO LINE (10,11)
0060 C
0061 0000 CONTINUE
0062 RETURN
0063 END

```

FTN4 COMPILER: HP92060-16092 REV. 2026 (000423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00002 COMMON = 00000

PAGE 8884 COOLN 12:26 PM WED.. 3 NOV.. 1982

```

0119 1# IF(IND256.LE.4#) GO TO 5
0120 IND256=IND256-4#
0121 ASSIGN 1# TO CODESV
0122 GO TO 1#8#
0123 15 CONTINUE
0124 C
0125 C CALCULATE MAKE UP CODE INDEX
0126 C
0127 INDEX=(LAM-B-1)*PMXD64*(AMB+PELMAX)/64+64+IND256
0128 2# IF(INDEX.LE.1#4) GO TO 3#
0129 INDEX=INDEX-4#
0130 DEX=1#4
0131 D ASSIGN 2# TO CODESV
0132 IF(DIAG)WRITE(TERM,25)CODESV,DEX,POLAR
0133 D25 FORMAT(1,2# CODESV,"316)
0134 GO TO 1#8#
0135 3# DEX=INDEX
0136 ASSIGN 4# TO CODESV
0137 IF(DIAG)WRITE(TERM,35)CODESV,DEX,POLAR
0138 D35 FORMAT(1,3# CODESV,"316)
0139 GO TO 1#8#
0140 C
0141 C CALCULATE TERMINATING CODE INDEX
0142 C
0143 4# DEX=MOD(AMB+PELMAX,64)+1
0144 ASSIGN 6# TO CODESV
0145 IF(DIAG)WRITE(TERM,45)CODESV,DEX,POLAR
0146 D45 FORMAT(1,4# CODESV,"316)
0147 GO TO 1#8#
0148 6# RETURN
0149 C
0150 C CODE LOOK-UP AND INSERTION ROUTINE
0151 C
0152 1#8# CONTINUE
0153 TCODE=CODE(3,DEX,POLAR)
0154 TLENG=CODE(1,DEX,POLAR)
0155 IF(DIAG) WRITE(TERM,1#8#) CODE(1,DEX,POLAR)
0156 D1#8#5 FORMAT(1,WORD LENGTH,"16)
0157 CALL M128(TCODE,CDBUF,CDELCT+1,TLENG)
0158 CDELCT=CDELCT+TLENG
0159 CDDATA=CDDATA+TLENG
0160 C
0161 GO TO CODESV,11#8,2#8,4#8,6#8)
0162 E N D

```

FTN4 COMPILER: HP9286B-16B92 REV. 2#26 (88B423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 88106 COMMON = 88888